LU12b: Datenschutz, Rollen & Zugriffe im Notenbuch

Lernziele dieser Seite

Sie können …

- erklären, warum Noten besonders geschützt werden müssen.
- typische Rollen im Schul-Alltag (Lernende, Lehrperson, Verwaltung, IT) benennen und deren Zugriffsrechte beschreiben.
- das Prinzip der minimalen Privilegien (Need-to-know) anhand eines Notenbuch-Beispiels erklären.
- den Zusammenhang zwischen Rollen im Alltag und Rollen/Rechten in einer Datenbank (Vorbereitung auf LU12c) verstehen.

1. Datenschutz und Noten – warum das wichtig ist

Noten sind **Personendaten**, die direkt einer identifizierten Person zugeordnet werden können (Name, Klasse, Lernende). Sie gehören zu den **besonders schützenswerten Daten**, weil sie:

- Aussagen über Leistung und Verhalten einer Person machen,
- Einfluss auf Laufbahnentscheidungen, Lehre, Studium und Beruf haben,
- bei Missbrauch zu Benachteiligung oder Diskriminierung führen können.

Deshalb verlangen **DSG (Schweiz)** und **DSGVO (EU)** unter anderem:

- Schutz vor unbefugtem Zugriff,
- Sicherstellung von Vertraulichkeit, Integrität und Verfügbarkeit,
- möglichst strikte Vergabe von Rechten: Nur wer Daten braucht, darf sie sehen oder bearbeiten.

Im Modul 290 setzen Sie diese rechtlichen Anforderungen technisch um – z. B. mit Benutzerkonten, Rollen und Berechtigungen in einer Noten-Datenbank.

2. Rollen und Zugriffe im Schul-Alltag

Bevor wir in die Technik gehen, überlegen wir: **Wer darf in der Schule was mit Noten tun?**

Typische Rollen:

- Lernende
- Lehrpersonen
- Verwaltung / Schulleitung
- IT-Verantwortliche

Überlegen Sie sich für das **Notenbuch-Tool**:

- Wer darf **Noten erfassen und ändern**?
- Wer darf **alle Noten einsehen**?
- Wer darf **gar keine Noten** sehen?

Eine mögliche Aufteilung:

Rolle	**Beispielperson**	**Typische Rechte im Notenbuch**
Lernende	z. B. Lernende der Klasse INF23a	sieht nur die eigenen Noten oder gar keine (je nach System)
Lehrperson	z. B. Guido Koch	erfasst und ändert Noten für „seine“ Klassen/Fächer
Verwaltung/Schulleitung	z. B. Studiengangsleitung	sieht alle Noten, kann Auswertungen erstellen
IT	z. B. System-Administrator	technisch Zugriff auf Datenbank, aber organisatorisch zur Verschwiegenheit verpflichtet

Wichtig:

- Nicht alle Personen brauchen die gleichen Rechte.
- Je sensibler die Daten, desto **enger** sollten die Rechte vergeben werden.

Diese Überlegungen bilden die **fachliche Grundlage**, die wir später in MySQL als Rollen und Berechtigungen abbilden.

3. Prinzip der minimalen Privilegien (Need-to-know)

Prinzip: Jede Person (oder Anwendung) erhält nur die Rechte, die sie für ihre Aufgabe unbedingt benötigt – **nicht mehr**.

Übertragen auf das Notenbuch:

- Lernende:
  - brauchen in der Regel **keine Rechte**, Noten direkt zu verändern,
  - können evtl. ihre eigenen Noten einsehen.
- Lehrpersonen:
  - brauchen Rechte, Noten für ihre Klassen **zu erfassen und zu ändern**,
  - müssen aber nicht zwingend alle Noten der ganzen Schule sehen.
- Verwaltung / Schulleitung:
  - braucht Zugriff auf viele oder alle Noten, z. B. für Promotionsentscheide,
  - sollte aber nicht willkürlich selbst Noten ändern.
- IT:
  - braucht technische Rechte (Backups, Updates),
  - sollte Noten zwar **schützen**, aber fachlich nicht bearbeiten.

Dieses Prinzip hilft:

- Risiken zu reduzieren (weniger Konten mit „Allmacht“),
- Fehler zu begrenzen (wer wenig darf, kann wenig kaputtmachen),
- DSG/DSGVO-Anforderungen zu erfüllen.

Auf der nächsten Seite sehen Sie dann, wie dieses Prinzip in MySQL mit Rollen wie `db_admin` und `lernende_role` umgesetzt wird.

4. Von Personenrollen zu IT-Rollen

In der Informatik sprechen wir auch von **Rollen**, meinen aber etwas Technisches:

- Eine Rolle in der Schule: z. B. *Lehrperson*, *Lernende*, *Verwaltung*.
- Eine Rolle in der IT / Datenbank: z. B. `db_admin`, `lernende_role`.

Die Idee ist gleich:

- Mehrere Personen können die **gleiche Rolle** haben.
- Einer Rolle sind bestimmte **Rechte** zugeordnet.

Beispiel Notenbuch:

- Fachliche Rolle: Lehrperson → Rechte: Noten erfassen/ändern für ihre Klassen.
- Technische DB-Rolle: `db_admin` → Rechte: alles auf `noten_db.*`.
- Technische DB-Rolle: `lernende_role` → Rechte: nur `SELECT` (Leserechte).

Die Schritte sind immer:

- Zuerst fachlich überlegen: **Wer soll was dürfen?**
- Dann technisch umsetzen: **Welche Rolle bekommt welche Rechte?**

5. Der root-User – mächtig, aber…

In jeder MySQL-Installation gibt es den **Benutzer `root`**. Er ist der **Superuser** – vergleichbar mit einem Systemadministrator auf einem Computer.

Eigenschaften von `root`:

- hat **alle Rechte** auf allen Datenbanken (`ALL PRIVILEGES ON *.*`),
- kann **andere Benutzer und Rollen anlegen, ändern oder löschen**,
- kann **jede Tabelle lesen, ändern oder entfernen**,
- ist **nicht beschränkt durch Rollen oder Privilegien**.

Das ist praktisch für **Systemverwaltung und Setup**, aber riskant im Alltag:

- Ein Fehler mit `root` kann **alle Datenbanken** betreffen (z. B. versehentliches Löschen),
- Wenn jemand das Passwort von `root` kennt, hat er **vollständigen Zugriff auf alle Daten**,
- Applikationen, die mit `root` verbunden sind, verletzen das Prinzip der **minimalen Privilegien**.

Best Practice:

- `root` wird **nur für administrative Aufgaben** genutzt (z. B. Setup, Benutzerverwaltung).
- Für den Betrieb (z. B. in Applikationen, Notenbuch, Übungsszenarien) werden **eigene Benutzer mit begrenzten Rechten** angelegt.
- Diese Benutzer erhalten nur die **Rollen und Rechte**, die sie wirklich brauchen.

6. Erste technische Begriffe

Auf der nächsten Seite lernen Sie die konkreten MySQL-Befehle kennen:

- `CREATE ROLE` – erstellt eine neue Rolle,
- `GRANT` – gibt einer Rolle oder einem Benutzer Rechte,
- `CREATE USER` – legt ein MySQL-Benutzerkonto an,
- `SET DEFAULT ROLE` – sorgt dafür, dass eine Rolle beim Login automatisch aktiv ist,
- `SHOW GRANTS` – zeigt, welche Rechte ein Benutzer effektiv hat.

Sie müssen diese Befehle auf dieser Seite hier noch nicht im Detail verstehen. Wichtig ist:

- Eine Rolle bündelt Rechte.
- Ein Benutzer bekommt eine oder mehrere Rollen.
- Über diese Kombination wird gesteuert, **wer was in der Noten-Datenbank tun darf**.