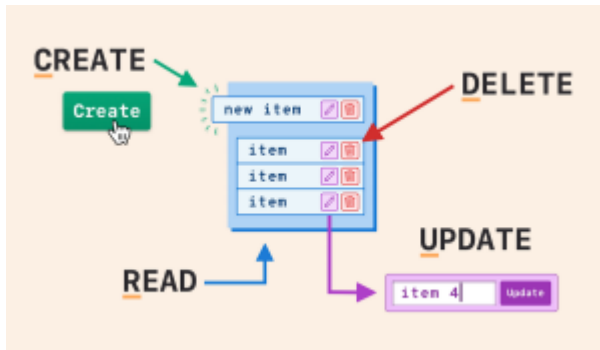


LU07a - CRUD: Die Basics der Datenverwaltung

CRUD steht für **Create, Read, Update, Delete** – die vier grundlegenden Operationen, mit denen jede Anwendung Daten **anlegt, liest, ändert, löscht**. CRUD ist überall: in Web-Apps, REST-APIs, Admin-Backends, Mobile Apps – und natürlich in SQL.



Lernziele

- Ich kann **CRUD** in eigenen Worten erklären.
- Ich kann CRUD-Operationen zu **REST-API** und **SQL** zuordnen.
- Ich verstehe, warum CRUD als **Skript** (Migration/Seed) oft besser ist als Klicks in der GUI.

CRUD in der Praxis

CRUD	Web/REST (Beispiel Zalando-Shop)	SQL (Datenbank-Ebene)
Create	POST /products	INSERT INTO products (name, price) VALUES ('Sneaker X', 120);
Read	GET /products?category=sneaker&minPrice=50	SELECT * FROM products WHERE category = 'sneaker' AND price >= 50;
Update	PATCH /products/42 { „price“: 99 }	UPDATE products SET price = 99 WHERE id = 42;
Delete	DELETE /products/42	DELETE FROM products WHERE id = 42;

—

Warum das wichtig ist

- **Gleiche Logik überall**

Ob im Webshop oder direkt in der Datenbank – es geht immer um CRUD.

Beispiel: Klick auf „nächste Seite“ → Browser macht einen ****GET****-Request

→ in der DB läuft ein ****SELECT**** mit ``LIMIT`/`OFFSET``.

- **Nachvollziehbar & versionierbar**

SQL-Skripte (Migrationen, Seeds) lassen sich in **Git speichern**. Damit ist klar: Wer hat wann welche Änderung gemacht?

GUI-Klicks (z. B. CSV-Import) verschwinden spurlos – niemand weiss später mehr, wie genau die Daten eingefügt wurden.

- **Schneller & sicherer**

Ein SQL-Skript kann **tausende Datensätze** in Sekunden einfügen, mit korrekten **Datentypen** und überprüften **Constraints**.

CSV-Importe per GUI sind fehleranfällig: Texte statt Zahlen, falsche Datumsformate, doppelte Einträge.

Mit Skripten ist das Ergebnis immer gleich – egal, auf welchem Rechner oder in welcher Umgebung.

Beispiele (Film-DB)

- **Create:** einen Film hinzufügen

```
INSERT INTO imdb_top_1000 (Series_Title, Released_Year,  
Genre, IMDB_Rating)  
VALUES ('Dune', 2021, 'Action, Adventure, Sci-Fi', 8.0);
```

- **Read:** Filme ab 2010 mit Top-Bewertung auslesen

```
SELECT Series_Title, Released_Year, IMDB_Rating  
FROM imdb_top_1000  
WHERE Released_Year >= 2010  
ORDER BY IMDB_Rating DESC;
```

- **Update:** Genre aktualisieren

```
UPDATE imdb_top_1000  
SET Genre = 'Drama, War'  
WHERE Series_Title = '1917' AND Released_Year = 2019;
```

- **Delete:** Tuple (Datensatz) entfernen

```
DELETE FROM imdb_top_1000  
WHERE Series_Title = 'Dummy Movie' AND Released_Year = 1900;
```

Merksatz

CRUD ist das „Alphabet“ der Datenarbeit. Egal ob API-Call oder SQL-Skript – am Ende passieren Create/Read/Update/Delete auf Tabellen.

From:
<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:
https://wiki.bzz.ch/modul/m290_guko/learningunits/lu07/theorie/a_crud?rev=1758659610

Last update: **2025/09/23 22:33**

