

# LU13d - Filter 2: JSONB

Aktuell nutzen wir den Datentyp JSONB, um die Metadatenzuspeichern.

```
CREATE TABLE IF NOT EXISTS vector_store (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    content TEXT NOT NULL,
    embedding vector(%d),
    metadata JSONB
);
```

JSONB (Binary JSON) erlaubt es JSON-Objekte in optimierter Form abzuspeichern und in den SQL-Statements die Einträge innerhalb des JSON-Objektes zu verwenden.

Das folgende Beispiel zeigt die erweiterte SQL-Schreibweise. Diese Abfrage kann in unserem Fall verwendet werden, um zu prüfen, ob bereits Seiten des Namespaces in der Datenbank gespeichert sind, ohne dass einzeln geprüft oder alle Datensätze zurückgegeben werden müssen.

```
SELECT metadata->>'source' AS SOURCE
  FROM vector_store
 WHERE metadata->>'source' LIKE ?
```

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



Permanent link:  
<https://wiki.bzz.ch/de/modul/ffit/3-jahr/java/learningunits/lu13/d?rev=1764637385>

Last update: **2025/12/02 02:03**