

G1G

Ich kann das Prinzip des Zugriffes bei einer NoSQL Datenbank erläutern. (z. B. Queries, Projections)

Fragenstellung und Lernziele

- Was versteht man unter einer Query in einer NoSQL Datenbank?

Erklären, wie Queries als Abfragebefehle genutzt werden, um gezielt Daten aus einer Datenbank zu filtern.

- Wie funktionieren Projections und welchen Zweck erfüllen sie?

Verstehen, wie Projections dazu verwendet werden, die zurückgegebenen Felder einer Query zu bestimmen, um die Ergebnismenge zu optimieren.

- Wie interagieren Queries und Projections miteinander?

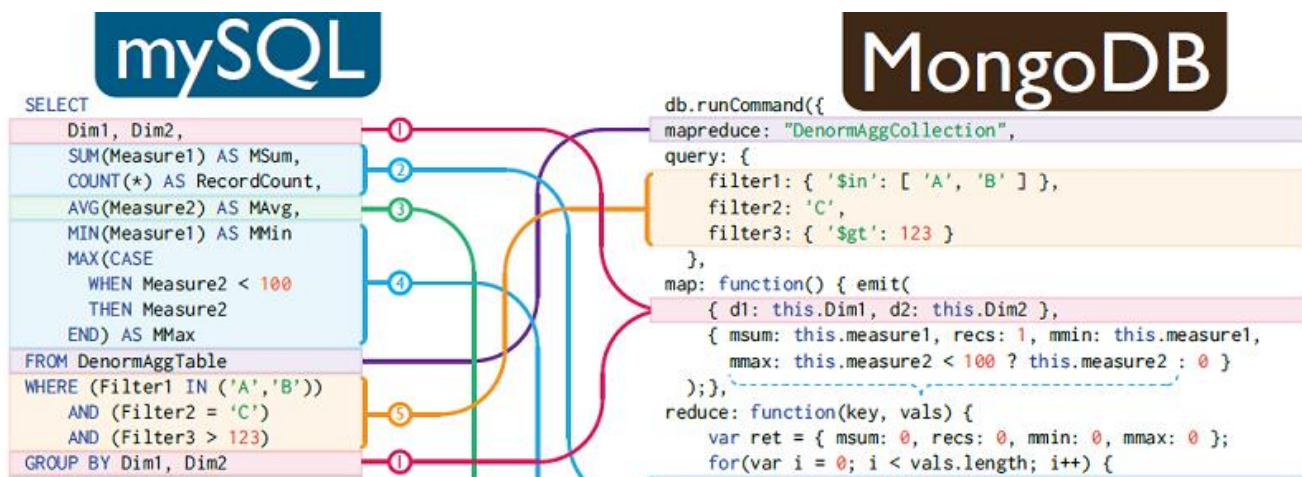
Erkennen, dass Queries zur Auswahl von Datensätzen dienen und Projections zur Steuerung des Inhalts der Ergebnisse.

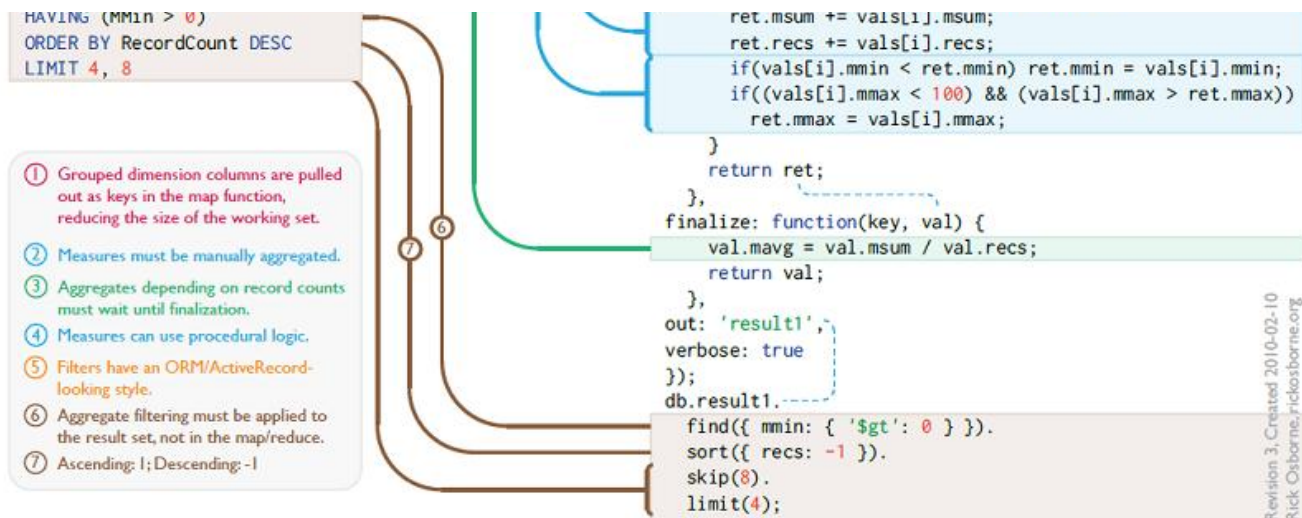
- Welche speziellen Operatoren und Syntaxelemente kommen in NoSQL-Abfragen zum Einsatz?

Die gängigen Operatoren (z. B. `$eq`, `$gt`, `$in`) und die JSON-basierte Abfragesyntax kennenlernen.

Umsetzung

In NoSQL Datenbanken wie MongoDB erfolgt der Datenzugriff primär über **Queries**. Diese ermöglichen es, mit Hilfe einer JSON-artigen Syntax gezielt Dokumente aus Collections auszuwählen. Ergänzend dazu kommen **Projections** zum Einsatz, um festzulegen, welche Felder in den Abfrageergebnissen enthalten sein sollen.





Queries in NoSQL Datenbanken

- **Definition und Funktion:** Eine Query ist ein Abfragebefehl, der Kriterien zur Auswahl von Dokumenten definiert.
- **Syntax und Operatoren:**
 - Abfragen werden in einer JSON-ähnlichen Struktur formuliert.
 - Gängige Operatoren sind beispielsweise:
 - `$eq`: prüft auf Gleichheit.
 - `$gt` / `$lt`: prüfen auf grösser oder kleiner als einen bestimmten Wert.
 - `$in`: prüft, ob ein Wert in einer Liste enthalten ist.
- **Beispiel einer Query:** Um alle Dokumente zu finden, in denen das Feld `status` den Wert `"active"` hat, kann folgende Query verwendet werden:

```
{
  "status": { "$eq": "active" }
}
```

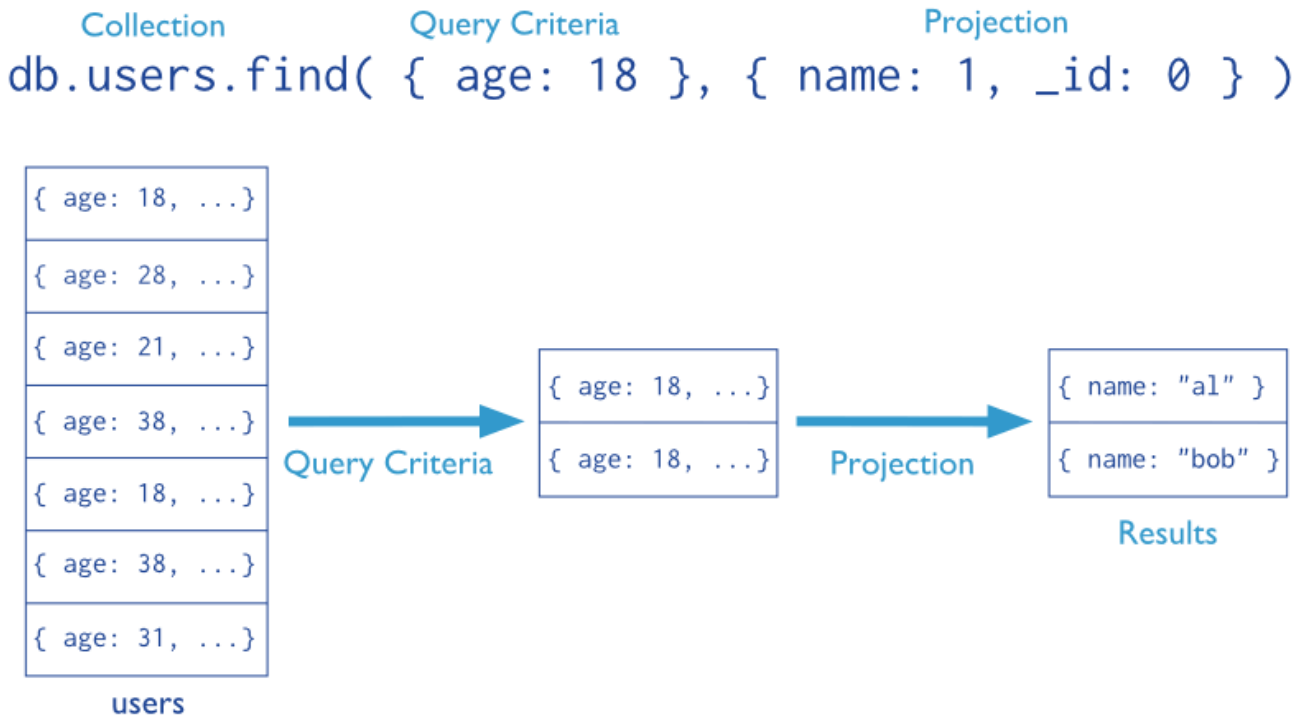
Dieses Beispiel demonstriert, wie mit einem einfachen JSON-Kriterium gezielt nach bestimmten Dokumenten gesucht wird.

Projections in NoSQL Datenbanken

- **Definition und Zweck:** Projections dienen dazu, die zurückgegebenen Felder einer Query zu bestimmen. Sie helfen, nur relevante Daten abzurufen und reduzieren somit die Menge der übertragenen Informationen.
- **Funktionsweise:**
 - Durch Angabe einer Projection kann man Felder ein- oder ausschliessen.
 - Typischerweise wird eine 1 für Inklusion und eine 0 für Exklusion verwendet.

- **Beispiel einer Projection:** Um bei der obigen Query nur die Felder `name` und `status` zurückzugeben, kann die Projection wie folgt ergänzt werden: `{ name: 1, "status": 1, "_id": 0 }`. Hierbei wird das `_id`-Feld explizit ausgeschlossen, während `name` und `status` in den Ergebnissen enthalten sind.

Zusammenspiel von Queries und Projections



- **Abfrage und Ergebnisoptimierung:** Zunächst filtert die Query anhand der definierten Kriterien die relevanten Dokumente heraus. Anschliessend wird mit der Projection bestimmt, welche Felder der gefundenen Dokumente im Ergebnis enthalten sein sollen.
- **Vorteile:** - **Effizienz:** Es werden nur die benötigten Daten abgerufen, was Netzwerk- und Verarbeitungsressourcen spart. - **Übersichtlichkeit:** Die Ergebnisse enthalten nur die für die Anwendung relevanten Informationen.
- **Kombination in der Praxis:** In einem typischen Anwendungsszenario könnte man eine Query mit einer zugehörigen Projection in einem einzigen Befehl kombinieren, um gezielt und effizient Daten zu extrahieren.

Operatoren und Syntaxelemente

Operatoren:

- `$eq`: Gleichheit
- `$gt`, `$lt`: Grösser oder kleiner als
- `$in`: Wert in Liste
- `$and`, `$or`: Logische Verknüpfungen

```
// Beispiel einer komplexen Query mit Projection
db.collection.find(
  {
    "status": { "$eq": "active" },
    "age": { "$gt": 18 },
    "hobbies": { "$in": ["reading", "traveling"] }
  },
  {
    "name": 1,
    "age": 1,
    "hobbies": 1,
    "_id": 0
  }
);
```

Wichtige Begriffe und Strukturen

- **Query:** Ein Abfragebefehl in JSON-Form, der Kriterien für die Auswahl von Dokumenten definiert.
- **Projection:** Eine Spezifikation, welche Felder in den Ergebnissen einer Query enthalten sein sollen.
- **Operatoren:** JSON-basierte Schlüssel wie `$eq`, `$gt`, `$in` usw., die zum Vergleich und zur Filterung von Daten verwendet werden.
- **JSON-Syntax:** Die standardisierte Form zur Darstellung von Daten, die in NoSQL-Abfragen genutzt wird.

Fazit

Das Zugriffsprinzip in NoSQL Datenbanken basiert auf einem flexiblen und leistungsfähigen Abfragesystem: - **Queries** ermöglichen die gezielte Auswahl von Dokumenten durch definierte Filterkriterien. - **Projections** steuern, welche Felder der abgefragten Dokumente zurückgegeben werden, wodurch die Effizienz und Übersichtlichkeit der Ergebnisse verbessert wird.

Dieses Prinzip erlaubt es, mit minimalem Aufwand präzise und ressourcenschonende Datenzugriffe in NoSQL Umgebungen zu realisieren.

Nachweis