

# MongoDB — agregacje i porównanie z SQL

Czas: 2 × 45 min

praca indywidualna lub w parach

**Cel:** Budowanie potoków agregacji w MongoDB (\$match, \$group, \$sort, \$project, \$unwind, \$lookup), porównanie ich z zapytaniami SQL na bazie sklep, użycie indeksu w MongoDB i ocena, kiedy wybrać model dokumentowy, a kiedy relacyjny.

## Po zajęciach potrafisz:

- buduje potok agregacji z etapami \$match, \$group, \$sort, \$limit i \$project
- rozpakowuje tablice etapem \$unwind i oblicza wartości wyrażeniami (\$multiply, \$sum, \$size)
- łączy kolekcje etapem \$lookup
- zakłada indeks i porównuje plan zapytania (COLLSCAN / IXSCAN) w explain
- porównuje te same raporty w SQL i MongoDB i dobiera model danych do zastosowania

## Sprzęt i materiały

- komputer z MongoDB Community Server i Compass (jak w zadaniu 28) oraz XAMPP z bazą sklep
- pliki `r2_28_produkty.json` i `r2_29_zamowienia.json` z folderu Pliki\_do\_zadan

## Oprogramowanie

- **MongoDB Compass** — zakładka Aggregations (kreator potoku) i konsola mongosh  
[mongodb.com/try/download/compass](https://mongodb.com/try/download/compass)
- **MongoDB Community Server** — lokalnie, tylko localhost  
[mongodb.com/try/download/community](https://mongodb.com/try/download/community)
- **XAMPP (MariaDB, phpMyAdmin)** — zapytania SQL na bazie sklep  
[apachefriends.org](https://www.apachefriends.org)

## Przygotowanie przed zajęciami

- Sprawdź, że baza sklep jest zaimportowana w XAMPP (plik sklep.sql z lekcji).
- Przypomnij sobie z lekcji 1.13–1.14 GROUP BY, HAVING i złączenia.

## Bezpieczeństwo

- Serwer MongoDB zostaw z domyślnym `bindIp: 127.0.0.1`. Nie włączaj dostępu z sieci, nawet po utworzeniu użytkownika.
- Nie włączaj w pracowni `security.authorization` w pliku `mongod.cfg` bez zgody nauczyciela — po restarcie usługi bez znanego hasła stracisz dostęp do bazy. Hasło użytkownika z Z12 wymyśl, nie wpisuj go na zrzutach ekranu.
- Przed zaimportowaniem kolekcji usuń jej starą wersję (z zadania 28) tylko w swojej bazie `sklep`.

## Przebieg zadania

### 1 Przygotowanie danych

W MongoDB jedno zamówienie to jeden dokument: dane klienta i tablica pozycji są w środku. W SQL te same dane są w czterech tabelach.

```
// fragment dokumentu z kolekcji zamowienia
{ _id: 1,
  klient: { id: 1, imie: "Anna", nazwisko: "Nowak", miasto: "Kraków" },
  data: ISODate("2026-02-10T09:15:00Z"), status: "wysłane",
  pozycje: [ { id_produktu: 1, produkt: "Laptop Vertex 14", ilosc: 1, cena: 3299 },
             { id_produktu: 7, produkt: "Mysz bezprzewodowa", ilosc: 1, cena: 129 } ] }
```

1. Z1. W bazie sklep (MongoDB) usuń kolekcję produkty z zadania 28 (`db.produkty.drop()`) i zaimportuj ją od nowa z `r2_28_produkty.json`. Utwórz kolekcję zamowienia i zaimportuj `r2_29_zamowienia.json`. Policz dokumenty w obu kolekcjach. Sprawdź w Compass, że pole data ma typ Date.

### 2 Potoki agregacji i ich odpowiedniki w SQL

`db.kolekcja.aggregate([ etap1, etap2, ... ])` — każdy etap przetwarza wynik poprzedniego. Do każdego potoku napisz zapytanie SQL na bazie sklep i porównaj wyniki. Wartość pozycji: `{ $multiply: ["$pozycje.ilosc", "$pozycje.cena"] }`.

1. Z2. Policz zamówienia o statusie „wysłane” (`$match`, `$count`).
2. Z3. Policz zamówienia w każdym statusie, od najliczniejszego (`$group`, `$sum: 1`, `$sort`).
3. Z4. Oblicz wartość każdego zamówienia i wyświetl 5 najdroższych (`$unwind`, `$group`, `$sort`, `$limit`).
4. Z5. Wyświetl 3 klientów (nazwisko) z największą łączną wartością zamówień, pomijając zamówienia anulowane.
5. Z6. Wyświetl 5 najczęściej kupowanych produktów (suma sztuk). Przy równej liczbie sztuk sortuj alfabetycznie.
6. Z7. Policz zamówienia w każdym miesiącu (`$dateToString` z formatem `"%Y-%m"`).
7. Z8. Dla każdego zamówienia wyświetl numer, nazwisko klienta, liczbę pozycji (`$size`) i numer miesiąca (`$month`) — bez `$unwind` (`$project`).
8. Z9. Oblicz przychód z każdej kategorii produktów (bez zamówień anulowanych). Kategoria jest w kolekcji produkty — połącz kolekcje etapem `$lookup` (`localField: "pozycje.id_produktu"`, `foreignField: "_id"`). Porównaj z SQL i wyjaśnij różnicę dla produktu Hub USB-C 7w1.
9. Z10. Znajdź produkty, które nie wystąpiły w żadnym zamówieniu (`$lookup` z kolekcji produkty do zamowienia i warunek na pustą tablicę).

### 3 Indeks i użytkownik

Plan zapytania w MongoDB wyświetla metoda `explain("executionStats")`: COLLSCAN oznacza przegląd całej kolekcji, IXSCAN — użycie indeksu.

1. Z11. Sprawdź plan zapytania o zamówienia klientów z Krakowa (pole klient.miasto): stage i totalDocsExamined. Załóż indeks na klient.miasto (`createIndex`), powtórz `explain` i porównaj. Wyświetl listę indeksów (`getIndexes`).
2. Z12. Utwórz użytkownika aplikacji `sklep_app` z rolą `readWrite` tylko do bazy sklep (`db.createUser`) i wyświetl użytkowników (`db.getUsers()`). Opisz w sprawozdaniu, co trzeba jeszcze zmienić w pliku `mongod.cfg`, żeby serwer wymagał logowania, i jak wtedy wygląda adres połączenia (`connection string`) — nie wykonuj tej zmiany.

#### 4 Kiedy SQL, a kiedy dokumenty

1. Z13. Uzupełnij w sprawozdaniu tabelę porównania: liczba poleceń/etapów potrzebnych do raportów Z4 i Z9 w SQL i MongoDB, oraz tabelę wyboru bazy dla podanych zastosowań z uzasadnieniem.

#### W sprawozdaniu umieść

- ☐ plik `zadanie29.js` z potokami Z1–Z12 i plik `zadanie29.sql` z odpowiednikami SQL (komentarze `// Z4, -- Z4`)
- ☐ zrzuty ekranu wyników Z4, Z9 (MongoDB i SQL) oraz obu planów z Z11
- ☐ wyjaśnienie różnicy wyników w Z9
- ☐ tabele porównania i wyboru bazy (Z13)
- ☐ odpowiedzi na pytania kontrolne

#### Pytania kontrolne

1. Do czego służy etap \$unwind i co by się stało w Z4 bez niego?
2. Który etap potoku odpowiada w SQL klauzuli WHERE, który GROUP BY, a który LEFT JOIN? Czy \$match po \$group odpowiada WHERE czy HAVING?
3. Dlaczego w MongoDB nazwisko klienta zapisano w dokumencie zamówienia, a w SQL tylko id\_klienta? Co się stanie po zmianie nazwiska klienta w obu modelach?
4. Podaj dwa zastosowania, w których wybierzesz MongoDB, i dwa, w których wybierzesz bazę relacyjną. Uzasadnij.

#### Kryteria oceny

| Kryterium                                                                   | Punkty    |
|-----------------------------------------------------------------------------|-----------|
| Z1–Z3: import danych i proste agregacje                                     | 2         |
| Z4–Z8: potoki z \$unwind, wyrażeniami i \$project wraz z odpowiednikami SQL | 7         |
| Z9–Z10: \$lookup i wyjaśnienie różnic z SQL                                 | 3         |
| Z11–Z12: indeks z analizą explain, użytkownik z minimalną rolą              | 3         |
| Z13 i odpowiedzi na pytania kontrolne                                       | 5         |
| <b>Razem</b>                                                                | <b>20</b> |

Sprawozdanie przygotuj na wzorze dołączonym do zadania (plik .docx o tym samym numerze).