

UNIVERSITY OF
WESTERN MACEDONIA



DEPARTMENT OF ELECTRICAL
& COMPUTER ENGINEERING

Εισαγωγή στη MongoDB

Προηγμένα θέματα βάσεων δεδομένων

Κατηγορίες Βάσεων Δεδομένων

- Σχεσιακές (Relational): Πίνακες με σχέσεις.
Π.χ. MySQL, PostgreSQL
- NoSQL:
 - Key-Value (π.χ. Redis)
 - Document (π.χ. MongoDB)
 - Columnar (π.χ. Cassandra)
 - Graph (π.χ. Neo4j)
- Time-Series: Ιδανικές για χρονοσειρές (π.χ. InfluxDB)
- Object-Oriented: Αποθήκευση ως αντικείμενα (π.χ. ObjectDB)

Στατιστικά Χρήσης Βάσεων Δεδομένων (2024)

- 43% χρησιμοποιούν μόνο RDBMS (π.χ. MySQL, PostgreSQL)
- 8% χρησιμοποιούν μόνο NoSQL (π.χ. MongoDB, Cassandra)
- 49% χρησιμοποιούν συνδυασμό RDBMS και NoSQL
- MongoDB: 5η σε παγκόσμια δημοτικότητα (db-engines.com)

Τι είναι η MongoDB;

Η MongoDB είναι μια **NoSQL βάση δεδομένων**, που σημαίνει ότι **δεν αποθηκεύει τα δεδομένα της σε πίνακες** όπως οι παραδοσιακές σχεσιακές βάσεις, αλλά σε **έγγραφα τύπου JSON ή BSON**.

Αυτά τα έγγραφα μοιάζουν με αντικείμενα JavaScript και είναι πολύ **ευέλικτα**, γιατί **δεν απαιτούν προκαθορισμένο σχήμα**. Δηλαδή, μπορούμε να προσθέτουμε ή να τροποποιούμε πεδία σε διαφορετικά έγγραφα χωρίς περιορισμούς.

Χάρη σε αυτό το χαρακτηριστικό, η MongoDB είναι **ιδανική για εφαρμογές που εξελίσσονται συνεχώς**, όπως **Web εφαρμογές** ή **Big Data** σενάρια, όπου ο όγκος και η μορφή των δεδομένων μπορεί να αλλάζει διαρκώς.

Επιπλέον, υποστηρίζει λειτουργίες όπως **ευρετήρια (indexes)** και **ερωτήματα (queries)** για γρήγορη ανάκτηση των δεδομένων, όπως θα δούμε και παρακάτω.

Ιστοσελίδα: <https://www.mongodb.com/>

Manual: <https://www.mongodb.com/docs/manual/>

Τι είναι Collection;

- Αντίστοιχο πίνακα στη MySQL
- Περιέχει έγγραφα
- Δεν απαιτείται schema
- Δημιουργείται αυτόματα με την πρώτη insert

Τι είναι Index;

Ένα **ευρετήριο (index)** είναι μια **δομή δεδομένων** που βοηθά τη MongoDB να **βρίσκει γρήγορα** τις εγγραφές (documents) που ικανοποιούν ένα ερώτημα.

Χωρίς ευρετήριο, η MongoDB πρέπει να διαβάσει **όλα τα έγγραφα της συλλογής** για να βρει αυτά που ταιριάζουν στο ερώτημα — αυτή η διαδικασία λέγεται **collection scan** και είναι **αργή**, ειδικά όταν υπάρχουν χιλιάδες ή εκατομμύρια έγγραφα.

Τι είναι Index;

- Ένα ευρετήριο (index) είναι μια δομή που επιταχύνει την αναζήτηση εγγράφων.
- Χωρίς index: η MongoDB διατρέχει όλη τη συλλογή (collection scan).
- Με index: η MongoDB βρίσκει άμεσα τις εγγραφές μέσω καταλόγου.

Παράδειγμα:

`db.Χρήστες.createIndex({ όνομα: 1 })` → ευρετήριο στο πεδίο 'όνομα'

Πλεονεκτήματα:

- ✓ Ταχύτερα queries
- ✓ Λιγότερο φορτίο στον επεξεργαστή
- ✓ Χρήσιμο σε joins & aggregations

Προσοχή:

- ⚠ Καταναλώνει μνήμη/χώρο
- ⚠ Πολλά ευρετήρια = πιο αργές εγγραφές

Τι είναι Aggregation Pipeline;

Η **Aggregation Pipeline** στη MongoDB είναι ένας τρόπος για να επεξεργαζόμαστε δεδομένα βήμα-βήμα, όπως θα κάναμε με **GROUP BY, WHERE, HAVING** κ.λπ. στη MySQL.

Αντί να γράφουμε μια απλή `find()`, μπορούμε να φτιάξουμε μια σειρά από στάδια (stages), όπου κάθε στάδιο **φιλτράρει, μετασχηματίζει ή ομαδοποιεί** τα δεδομένα.

Μοιάζει με μια **γραμμή παραγωγής**: κάθε στάδιο παίρνει τα δεδομένα από το προηγούμενο, τα αλλάζει με κάποιον τρόπο και τα περνάει στο επόμενο.

Τι είναι Aggregation Pipeline;

javascript

```
db.Χρήστες.aggregate([
  { $match: { ενεργός: true } }, // στάδιο 1: φίλτρο
  { $group: { _id: "$πόλη", μέσος_όρος: { $avg: "$ηλικία" } } } // στάδιο 2: ομαδοποίηση
])
```

Εδώ κάνουμε δύο πράγματα:

1. Πρώτα, κρατάμε μόνο τους ενεργούς χρήστες με \$match
2. Μετά, τους ομαδοποιούμε ανά πόλη και υπολογίζουμε τη μέση ηλικία με \$group και \$avg.

Η Aggregation Pipeline είναι ιδανική για αναφορές, στατιστικά, dashboards και γενικά για περιπτώσεις όπου χρειαζόμαστε συγκεντρωτικά αποτελέσματα.

MongoDB vs MySQL

MONGO

- ✓ NoSQL, documents
- ✓ Χωρίς schema
- ✓ Επερωτήσεις με JavaScript

σύνταξη

MYSQL

- ✓ Σχεσιακή, πίνακες
- ✓ Προκαθορισμένο schema
- ✓ SQL queries

INSERT / QUERY - MySQL

MYSQL

- CREATE TABLE Χρήστες (...)
- INSERT INTO Χρήστες (...)
- SELECT * FROM Χρήστες WHERE ...

MONGODB

- db.Χρήστες.insertOne({...})
- db.Χρήστες.find({...})

Query σε MySQL και MongoDB

MYSQL

```
SELECT * FROM Χρήστες  
WHERE ηλικία > 25 AND ενεργός =  
TRUE;
```

MONGO DB

```
db.Χρήστες.find({  
  ηλικία: { $gt: 25 },  
  ενεργός: true  
});
```

Και τα δύο παραδείγματα επιστρέφουν χρήστες με ηλικία πάνω από 25 που είναι ενεργοί.

- Η MySQL χρησιμοποιεί SQL σύνταξη.
- Η MongoDB χρησιμοποιεί JSON-style σύνταξη με τελεστές όπως \$gt (greater than).

Συμπεράσματα

- MongoDB για δυναμικές εφαρμογές και μεγάλα δεδομένα
- MySQL για σταθερά, καλά ορισμένα σχήματα
- Επιλογή ανάλογα με τις ανάγκες του έργου