



Εργαστηριακό μάθημα 6

-

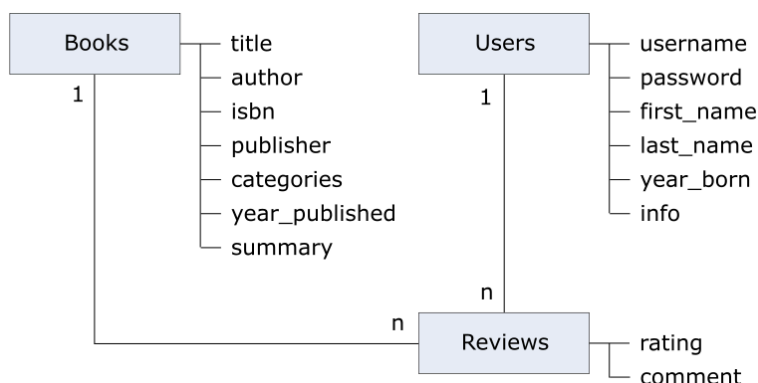
MongoDB

Ανάπτυξη NoSQL Βάσης Δεδομένων

Εισαγωγή

Θα υλοποιήσουμε την *library*, μια απλή ΒΔ που θα διατηρεί πληροφορίες για βιβλία και που θα επιτρέπει σε χρήστες να γράφουν reviews για αυτά.

Παρακάτω δίνεται το σχήμα της βάσης, το οποίο περιλαμβάνει τις συλλογές (collections) και τη δομή αυτών (fields). Επιπλέον δίνεται η πραγματική συσχέτιση μεταξύ των collections (χωρίς να μπορεί να εφαρμοστεί με χρήση ξένων κλειδιών - αναφορική ακεραιότητα).



Σύνδεση στη τοπική MongoDB

Ανοίξτε το MongoDB Compass. Επιλέξτε το local instance του Mongo server και συνδεθείτε.

Δημιουργία της ΒΔ

Δημιουργούμε μια νέα database επιλέγοντας το κουπί Create Database. Ονομάζουμε τη βάση *library* και ως πρώτο collection βάζουμε το **books** (στη mongo απαιτείται κάθε βάση να έχει τουλάχιστον ένα collection).



Create Database

Database Name

Collection Name

☐ Capped Collection ⓘ
☐ Use Custom Collation ⓘ

Before MongoDB can save your new database, a collection name must also be specified at the time of creation. [More Information](#)

CANCEL

CREATE DATABASE

Δημιουργία ΒΔ σε Atlas

Για να δημιουργήσουμε τη ΒΔ με τη χρήση του MongoDB Atlas, πρέπει πρώτα από όλα να δημιουργήσουμε έναν λογαριασμό: <https://www.mongodb.com/cloud/atlas/register>

Κατόπιν, θα βρεθείτε στην αρχική σας σελίδα, όπου θα πρέπει να πάτε στην επιλογή Clusters (κάτω από το Database), κατόπιν στο Collections και από εκεί να πατήσετε την επιλογή Create Database, όπως βλέπετε παρακάτω:

Create Database

Database name ⓘ

Collection name ⓘ

Additional Preferences

Capped Collection ⓘ

Size

Cancel

Create

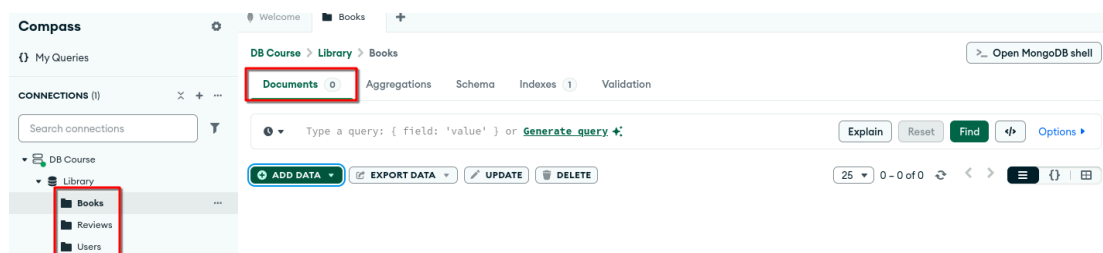


Δημιουργία των Collections της ΒΔ

Επιλέγουμε τη βάση **library** που δημιουργήσαμε και στη συνέχεια πατάμε Create Collection για να φτιάξουμε τις συλλογές της. Η συλλογή books είναι ήδη έτοιμη, οπότε στη συνέχεια φτιάχνουμε τις συλλογές **users** και **reviews** όπως παρακάτω.

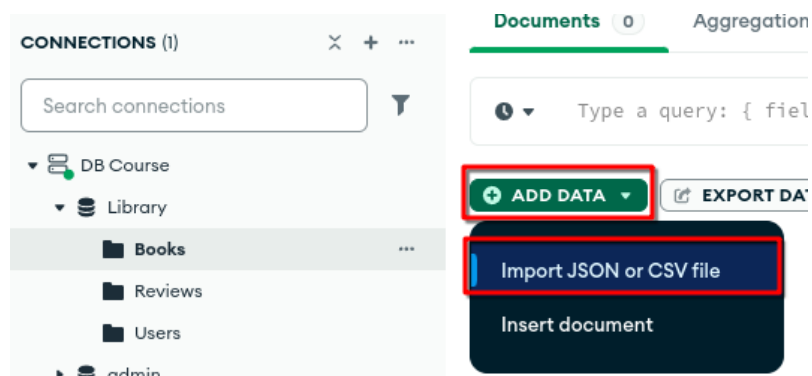
Εισαγωγή Δεδομένων στη ΒΔ

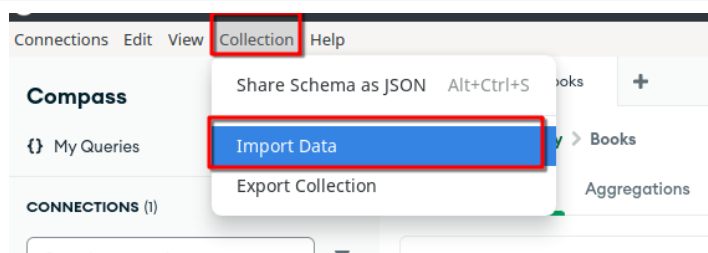
Μπορείτε να δείτε και να επιλέξετε Collections της ΒΔ από το αριστερό μενού στο εργαλείο MongoDB Compass.



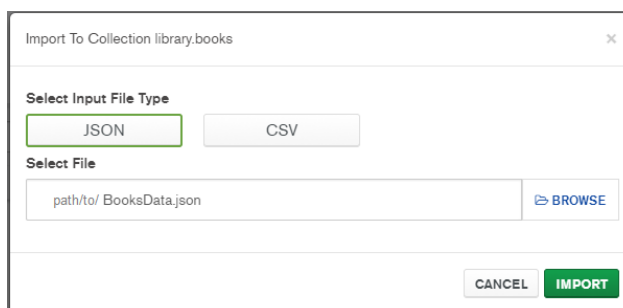
Για να εισάγουμε ένα έγγραφο (document) σε κάποιο collection, από το μενού επιλέγουμε **Add Data -> Insert Document**.

Εναλλακτικά, μπορούμε να προσθέσουμε δεδομένα από αρχεία μορφής JSON ή CSV. Για εισαγωγή των δεδομένων στη συλλογή books από το αρχείο “BooksData.json”, επιλέγουμε από το μενού **Add Data -> Import JSON or CSV File**. Εναλλακτικά από το πάνω μενού της εφαρμογής επιλέγουμε **Collection -> Import Data**.





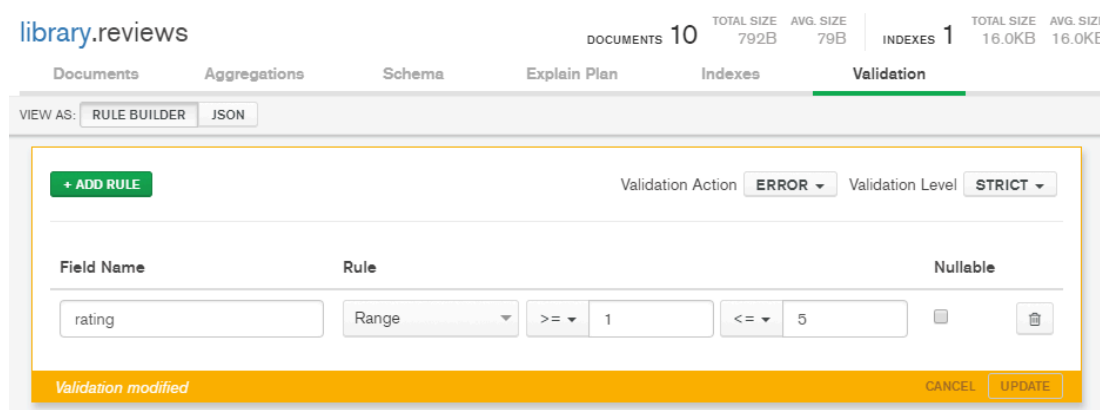
Στη συνέχεια, στο παράθυρο που ανοίγει, επιλέγουμε και φορτώνουμε το αρχείο BooksData.json από τον δίσκο.



Εκτελώντας την ίδια διαδικασία φορτώνουμε τα δεδομένων των Users και Reviews από τα αντίστοιχα αρχεία “UsersData.json” και “ReviewsData.json”

Περιορισμοί - Rules

Παρόλο που δε μπορούμε να εισάγουμε περιορισμούς αναφορικής ακεραιότητας (ξένα κλειδιά), είναι εφικτό να εισάγουμε κάποιους περιορισμούς για τα πεδία των εγγράφων. Για παράδειγμα, για τη συλλογή **reviews** μπορούμε από την καρτέλα **Validation** να προσθέσουμε έναν κανόνα σύμφωνα με τον οποίο κάθε rating πρέπει να είναι στο διάστημα 1 - 5.



Επιλέγοντας validation action ίσο με Warning, η βάση θα επιτρέψει την αλλαγή (δημιουργία ή ενημέρωση) εγγράφων που δεν πληρούν τον περιορισμό αλλά θα μας προειδοποιήσει. Αν το validation action είναι Error, τότε η βάση δε θα επιτρέψει αλλαγές που δεν πληρούν τον περιορισμό.



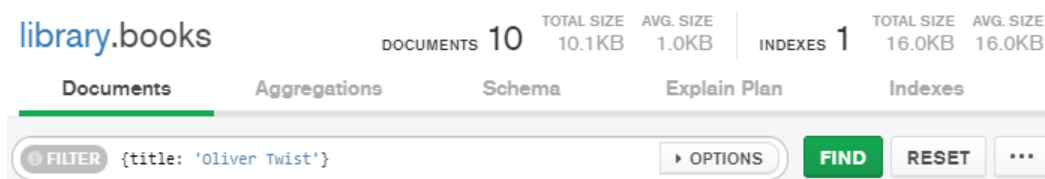
Με το validation level επιλέγεται το που θα εφαρμοστεί ο περιορισμός. Μπορεί να μην εφαρμοστεί καθόλου (Off), να εφαρμοστεί μόνο κατά την εισαγωγή/ενημέρωση εγγράφων που πληρούν τα κριτήρια (Moderate), ή να εφαρμοστεί για όλες τις εισαγωγές/ενημερώσεις (Strict). Πρακτικά, με το Moderate, ενημερώσεις σε έγγραφα που δεν πληρούν τα κριτήρια δεν ελέγχονται για την εγκυρότητά τους.

Σημείωση: Σε εκδόσεις του MongoDB Compass που δεν περιλαμβάνουν rule builder, μπορούμε να καταγράψουμε τον περιορισμό όπως παρακάτω:

```
{ rating: { $gte: 1, $lte: 5 } }
```

Δημιουργία Ερωτημάτων Filter

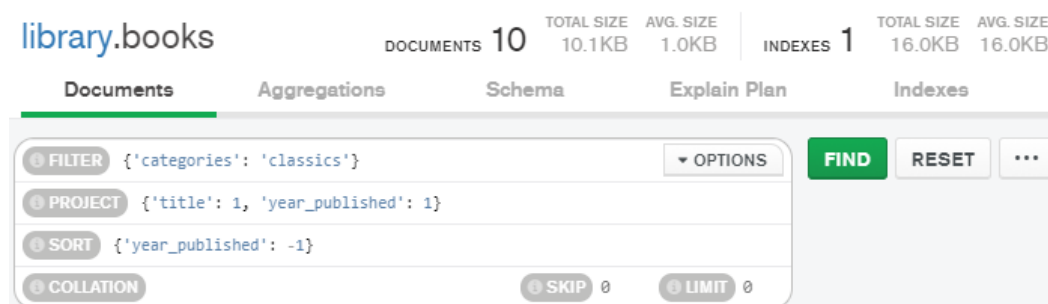
Μπορούμε να εκτελέσουμε κάποια ερωτήματα για να επιλέξουμε έγγραφα στο πεδίο filter. Για παράδειγμα, το παρακάτω ερώτημα θα επιστρέψει τα βιβλία με τίτλο “Oliver Twist”.



Ένα άλλο πιθανό ερώτημα αν θέλαμε να βρούμε όλα τα έργα του Dickens είναι το ερώτημα

```
{"author.last_name": "Dickens"}
```

Μπορούμε επίσης να εκτελέσουμε πολύπλοκα ερωτήματα filter, όπως π.χ. το παρακάτω:



Στο παραπάνω ερώτημα στο πεδίο filter βρίσκουμε όλα τα έργα που είναι στην κατηγορία classics. Για τα έργα αυτά στο πεδίο project επιλέγουμε να εμφανίσουμε τον τίτλο (title) και τη χρονολογία τους (year_published). Επίσης τα ταξινομούμε με το πεδίο sort (1 για αύξουσα σειρά, -1 για φθίνουσα)· η ταξινόμηση γίνεται με βάση τη χρονολογία σε φθίνουσα σειρά.



Κατασκευή Ευρετηρίου

Για κάθε ερώτημα, μπορούμε επιπλέον να δούμε το κόστος εκτέλεσής του επιλέγοντας το **Explain Plan**, καθώς επίσης και να φτιάξουμε κάποιο ευρετήριο για να επιταχύνουμε τα σχετικά ερωτήματα.

DB Course > Library > Books

Documents 10 Aggregations Schema Indexes 1 Validation

{'author.last_name': 'Dickens'}

Generate query Explain Reset Find

Στο ερώτημα `{title: 'Oliver Twist'}` για τον τίτλο ενός βιβλίου χρησιμοποιώντας το Explain Plan προκύπτει ότι προσπελούνται 10 έγγραφα:

Query Performance Summary

Documents Returned: 1	Actual Query Execution Time (ms): 0
Index Keys Examined: 0	Sorted in Memory: no
Documents Examined: 10	No index available for this query.

Κατασκευάσουμε ένα ευρετήριο στον τίτλο του βιβλίου από την καρτέλα **Indexes**

DB Course > Library > Books

Documents 10 Aggregations Schema Indexes 1 Validation

Create Refresh

VIEWING INDEXES

	Size	Usage	Properties	Status
Search Index	20.5 KB	8 (since Tue Dec 03 2024)	UNIQUE	READY

Create Index

Choose an index name

TitleIndex

Configure the index definition

title 1 (asc)

ADD ANOTHER FIELD

> Options

CANCEL CREATE

Στη συνέχεια, επιλέγοντας πάλι να αναλύσουμε το πλάνο εκτέλεσης τους ερωτήματος, προκύπτει ότι προσπελάσσεται 1 έγγραφο (σε αντίθεση με τις 10 προσπελάσεις χωρίς index):



Query Performance Summary

Documents Returned: 1

Index Keys Examined: 1

Documents Examined: 1

Actual Query Execution Time (ms): 0

Sorted in Memory: no

Query used the following index: title ↑

Δημιουργία Charts

Στο MongoDB Atlas μπορούμε να χρησιμοποιήσουμε το εργαλείο Chart Builder για να δημιουργήσουμε dashboards για τη ΒΔ μας.

Επιλέγουμε το Charts για να μεταβούμε στο συγκεκριμένο παράθυρο δημιουργίας των γραφημάτων, και στη συνέχεια επιλέγουμε Add Dashboard.

The screenshot shows the MongoDB Atlas interface. At the top, there's a navigation bar with 'Atlas', 'Konstantino...', 'Access Manager', and 'Billing'. Below this, the 'DB Course' is selected, and the 'Data Services' menu is open, with 'Charts' highlighted. The main content area shows 'Cluster0' with tabs for 'Overview', 'Real Time', 'Metrics', and 'Collections'. Below this, the 'Dashboards' section is visible, showing a list of dashboards including 'Restaurants' and 'Konstantinos's Dashboard'. The 'Add Dashboard' button is highlighted in the top right of the dashboards list.

Θα δημιουργήσουμε ένα Chart για την οπτικοποίηση της βαθμολογίας των εστιατορίων σε συνάρτηση με τον αριθμό των εστιατορίων.

The screenshot shows the 'Add Chart' dialog in the MongoDB Atlas Chart Builder. The 'Data Source' section is active, showing 'Select' and 'Library' options. The 'Chart Type' is set to 'Grouped Bar'. The 'Preview' section shows a visualization of the data. The 'Add Chart' button is highlighted in the top right of the dialog.



Πρέπει να επιλέξουμε την πηγή δεδομένων (data source). Για το συγκεκριμένο ερώτημα, θα επιλέξουμε ένα από τα έτοιμα collections που έχει η MongoDB Atlas, και θα διαλέξουμε το *sample_restaurants* και μετά το *restaurants*. Προφανώς μπορούμε να φτιάξουμε το διάγραμμα με δικά μας δεδομένα, αλλά εν προκειμένω, έχει πολλά documents, οπότε θα βγει πιο όμορφο αποτέλεσμα!

Select Data Source



Suggested **Project** Sample

▼ Cluster0 (Cluster)

- ▶ library
- ▶ sample_airbnb
- ▶ sample_analytics
- ▶ sample_geospatial
- ▶ sample_guides
- ▶ sample_mflix
- ▶ sample_restaurants
- ▶ sample_supplies
- ▶ sample_training
- ▶ sample_weatherdata

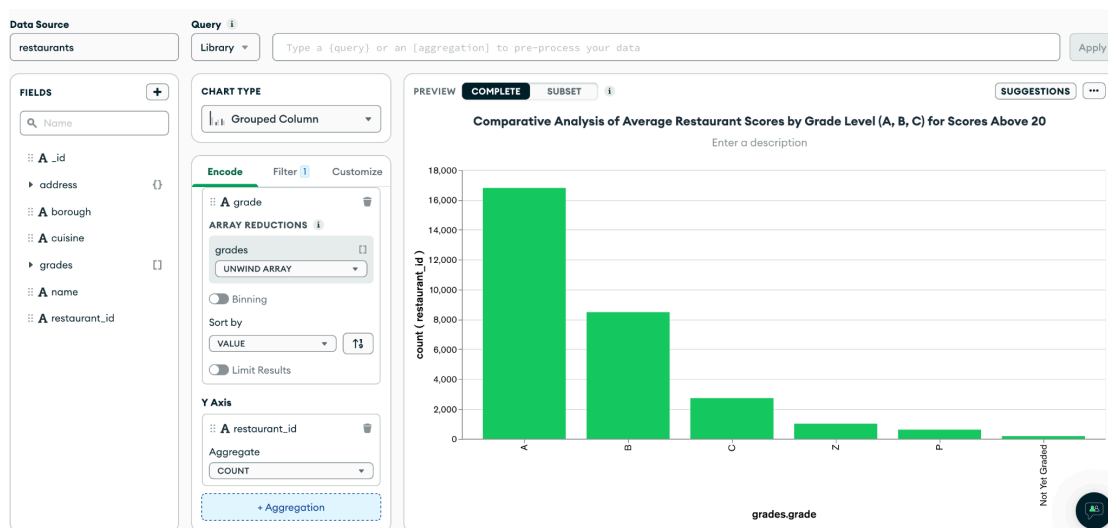
Για να δημιουργήσετε το διάγραμμα, πρέπει να κάνετε drag and drop από τον τομέα Fields τα πεδία που σας ενδιαφέρουν να απεικονιστούν, στον αντίστοιχο άξονα. Για να βάλουμε τους βαθμούς στον X άξονα, πρέπει από το πεδίο grades, να επιλέξουμε το grade, και να επιλέξουμε τις επιλογές που φαίνονται στην παρακάτω εικόνα.

Για να έχουμε το σύνολο των εστιατορίων, πρέπει να σύρουμε είτε το name, είτε το restaurant_id, ώστε να πάρει όλες τις καταχωρήσεις.

Φυσικά, μπορείτε να κάνετε άλλα διαγράμματα, ανάλογα με το τι θέλετε να οπτικοποιήσετε. Εδώ πχ μπορείτε να χρησιμοποιήσετε και Grouped Bar (από το μενού chart type).



Το τελικό αποτέλεσμα φαίνεται στην παρακάτω εικόνα:



Σημείωση

Μπορούμε να φιλτράρουμε τα έγγραφα με βάση εμφωλευμένα fields, όπως το score κριτικής των χρηστών για τα εστιατόρια, χρησιμοποιώντας ειδικούς selectors (περισσότερα στο επόμενο εργαστήριο) - εδώ χρησιμοποιούμε το φίλτρο να έχουν σκορ πάνω από 20

```
{grades: { $elemMatch: { score: { $lt: 2 } } } }
```