NSQ1 S25 Course Assignment 2

# Question 1 – Revise model

If necessary, revise the model from course assignment 1. Otherwise, use the same model.

# Question 2 – Model database

Design a MongoDB model for the bookstore model from the 1st course assignment. Use design patterns where appropriate and beware the anti-patterns. Make note of the choices you make and why. Document the model in one of the 4 ways shown in the slides.

Design the model to be used with the queries from the 1st course assignment.

# Question 3 – Create schemas and indexes

Design and create schemas for the collections in your model. Design indexes where it seems appropriate.

# Question 4 – Work with data

Answer the following questions in MongoDB using your model from question 1.

## 4a. Modifying data

Use MongoDB CRUD methods to execute the following scenarios. Use transactions where necessary.

If nothing else is stated, assume you know the object ids of the objects involved.

1. Sell a book to a customer.
2. Change the address of a customer.
3. Add an author to a book.
4. Retire the "Space Opera" category and assign all books from that category to the parent category. Don't assume you know the id of the parent category.
5. Sell 3 copies of one book and 2 of another in a single order

## 4b. Querying data

Write find() or aggregate() statements to return the following data

1. All books by an author
2. Total price of an order
3. Total sales (in £) to a customer
4. Books that are categorized as neither science fiction nor fantasy (**Note:** This is harder than you think. A book can have more categories. Make sure you don’t return books that are fantasy romance, for instance.)
5. Average page count by genre
6. Categories that have no sub-categories
7. ISBN numbers of books with more than one author
8. ISBN numbers of books that sold at least X copies (you decide the value for X)
9. Number of copies of each book sold – unsold books should show as 0 sold copies.
10. Best-selling books: The top 10 selling books ordered in descending order by number of sales.
11. Best-selling genres: The top 3 selling genres ordered in descending order by number of sales.

**Note:** In the next three exercises, subcategories of science fiction also count as science fiction, and so do subcategories of subcategories, and so on. It’s not enough to just look for the science fiction category.

1. All science fiction books. Note: Books in science fiction subcategories like cyberpunk also count as science fiction. Don’t use your knowledge of the concrete category structure. (Depending on the model you might want to use $graphLookup)
2. Characters used in science fiction books.
3. For each category: Number of books in the category including books in its subcategories.

# Question 5 – Cloud

1. Create a cluster and a database in MongoDB Atlas. Create a collection for each of the collections in Question 2.
2. Create a new collection, orderLog, and a trigger on creation of orders. The trigger should add objects to the orderLog whenever an order is inserted or updated.
   1. When inserted: The object should have a single property, “inserted”, with the order object as a value.
   2. When updated: The object should have an “updated” property with the new order, and an “updateDescription” property with the [update description](https://www.mongodb.com/docs/manual/reference/change-events/update/).
3. Implement the indexes from question 3.

Document this with

1. The code of the trigger function
2. Screenshots of the implemented indexes

# Question 6 – GraphQL

1. Create a schema definition in GraphQL Definition Language for enough your data model to cover the queries and mutations below.
2. Define queries in the Query type:
   * Given a search term, all books that have the search term as part of the title
   * Given the email of a customer, all orders from the customer with number of books and total prices
3. Define mutations in the Mutation type:
   * Create an order for a single book and a customer (1-click)
   * Apply x% reduction on a book

# Question 7 – Report

Write a report on the experience gained by completing Question 1 and 2 above. The report should contain answers to the questions

* What were the decisions taken in the modelling?
* Why were these decisions taken?
* What were the consequences of these decisions?
* What were the difficult and easy parts of the exercise?
* How does that compare to relational databases?
* What are the advantages and disadvantages of MongoDB over relational databases for this exercise?

# Rules

* Make the exercise in groups of 2 – 4
* Hand in to itslearning no later than 23 March