Universidad Complutense de Madrid x NTIC
Manuel Sanchez Mc Sweeney
NoSQL

Project Report: NoSQL Database Final Practice

1. Introduction and Objective

This document summarizes the final practical exercise for the NoSQL Databases module. The core objective of this project was to demonstrate proficiency in performing a wide range of database operations on a NoSQL database, specifically **MongoDB**. The practice involved setting up a database, importing a JSON dataset, and executing a series of queries and manipulations to analyze and modify the data. The entire process was conducted using the MongoDB shell.

2. Environment and Dataset

- **Database System:** The project was implemented using **MongoDB**, a popular document-oriented NoSQL database.
- **Dataset:** The dataset used was **`zara_us_sample_data.json`**, a JSON file containing product information from Zara's US catalog. Each document in the dataset represents a product and includes fields such as **`name`, `price`, `colors`, `is_new`**, and detailed size information.
- **Setup:** The first step involved setting up the database environment. A new database named **`Zara`** was created, and the contents of the **`zara_us_sample_data.json`** file were imported into a collection named **`products`** using the **`mongoimport`** command.

3. Database Operations and Queries

The practical work was divided into several queries and data manipulation tasks to showcase a comprehensive understanding of MongoDB's functionalities.

3.1. Basic Data Analysis & Retrieval

- **Counting Documents:** Basic queries were run to explore the dataset, such as counting the total number of products in the collection.
- **Conditional Queries:** Executed **`find()`** operations with specific criteria to filter data, for instance, retrieving all products marked as new **`(is_new: true)`**.

3.2. Price and Color Analysis

- **Price Extremes:** The most expensive and cheapest products were identified by using the **`sort()`** method on the **`price`** field in both descending and ascending order, combined with **`limit(1)`** to get the top result.

- **Average Price:** The aggregation framework was used to calculate the average price of all products. This was achieved with a `$group` stage and the `$avg` accumulator operator.
- **Color Analysis:**
    - The `distinct()` method was used to find all unique product colors available in the dataset.
    - An aggregation pipeline was constructed to count the number of products available for each color, demonstrating the use of `$unwind`, `$group`, and `$sum`.

3.3. Data Manipulation (Update and Delete)

- **Adding New Fields:** The `updateMany()` command was used to add a new boolean field, `on_sale`, with a default value of `false` to all documents in the collection.
- **Conditional Updates:** A more specific update was performed to set `on_sale` to `true` for all products with a price lower than $50, showcasing the use of query selectors (`$lt`) within an update operation.
- **Data Deletion:** The `deleteMany()` command was used to remove all documents that met a specific condition, in this case, all products where `is_new` was `true`.

3.4. Performance and Advanced Operations

- **Indexing:** To optimize query performance, an ascending index was created on the `price` field using the `createIndex()` method. This is a crucial step for improving the speed of sorting and querying on that field.
- **Advanced Aggregation (`$facet`):** A complex query was built using the `$facet` aggregation stage to perform multiple, independent aggregations in a single command. This was used to simultaneously categorize products into price ranges and provide a count of products per color.

4. Conclusion

This project successfully demonstrated the core competencies required for working with a NoSQL database like MongoDB. The tasks completed spanned the full lifecycle of data management, from initial data import and basic querying to complex aggregations, data modification, and performance optimization through indexing. The work reflects a solid practical understanding of both basic and advanced MongoDB operations.