Portfolio (Week-2)

This week, we focused on learning about **MongoDB**, a NoSQL database used for storing and managing data in modern web applications. The task for this lab involved working with a CSV file named people.csv. Below is a detailed account of how I approached this week’s lab:

1. **Downloading and Setting Up the File** First, I downloaded the people.csv file from Moodle. After downloading, I saved it in my Week 2 folder, which I had organized specifically for this module. This ensured all related files were easy to access and manage.
2. **Creating the Database** I began by creating a database in MongoDB. This involved:
   * Opening MongoDB and navigating to the appropriate folder.
   * Importing the people.csv file into the database. This process converted the CSV data into a structured format suitable for querying and manipulation in MongoDB.

A screenshot of a computer

Description automatically generated

1. Updating **Records** Once the file was successfully added to the database, I explored how to update records:

* By clicking the pencil icon next to a record, I could modify the values directly.
* After making the changes, I clicked the **Update** button to save my edits. This made it easy to correct or enhance the data in real-time.

A white screen with red dots

Description automatically generated

1. **Deleting Records** For deleting records, I simply selected the items I wanted to remove and clicked on the delete icon. This instantly removed the selected entries from the database.

A white background with red text

Description automatically generated

1. **Aggregation Operations** MongoDB’s aggregation framework allows for advanced data manipulation and analysis. To perform aggregations:

* I navigated to the **Aggregation Tab**.
* Created stages to apply different operations like matching, grouping, and projecting fields to analyse the data.

Some of the specific tasks I performed include:

* **Finding Education for Master’s Degree Holders:** Repeating the aggregation process to search for education fields containing “Master” and extracting relevant results.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Finding Average, Min, and Max Age and Salary by Marital Status:** Using grouping operations, I calculated the average, minimum, and maximum age and salary of people, categorized by their marital status.

A screenshot of a computer

Description automatically generated

**Analyzing Salary by Gender and Age Group:** I grouped the data by age and gender to find the minimum, maximum, and average salary for each group. Separate queries were run for males and females.

A screenshot of a computer

Description automatically generated

**Counting Married and Unmarried Individuals:** Using MongoDB’s aggregation features, I counted the number of married and unmarried males and females in the dataset. This required careful use of filters and grouping operations.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

To Count married and unmarried females and males:

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

**Learning Objectives**

* To understand how to import and manage data in MongoDB.
* To practice updating and deleting records in a database.
* To explore and apply MongoDB’s aggregation framework for data analysis.
* To learn how to organize and query data efficiently.

**Key Points**

* MongoDB is a powerful tool for managing and analyzing large datasets.
* Aggregation stages provide a systematic way to perform complex data analysis tasks.
* Proper data organization and file management simplify the workflow.

**Problems Faced**

1. **Uncertainty with Aggregation Queries:** Initially, I was unsure how to construct the queries to count and group data effectively. I struggled with matching fields and applying filters correctly.
   * **Solution:** I referred to the MongoDB documentation and examples provided in class. With these resources, I was able to experiment and eventually succeed in constructing the correct queries.
2. **Error Importing CSV File:** While importing the people.csv file, I encountered errors due to incorrect formatting in the file.
   * **Solution:** I opened the CSV file in a text editor, identified formatting issues (such as extra spaces or missing headers), and corrected them before re-importing.
3. **Misunderstanding Gender-Based Analysis:** While calculating salary statistics by gender, I accidentally included entries where gender data was missing.
   * **Solution:** I revised the query to exclude rows with null or undefined gender fields, ensuring accurate results.

By overcoming these challenges, I gained a deeper understanding of MongoDB’s features and capabilities. This week’s lab helped build a strong foundation for database management and querying.