**Assignment 01**

The chosen dataset isthe *Youtube Videos Dataset (~3400 videos)*. I decided to use this dataset for the assignment because it is one of the datasets I want to use and explore in my final project. I want to test the water with this dataset to see the potential of my project and if I’ll need to adjust and make some changes accordingly in this artistic process. I faced several issues during the package installation and data importation process. It was a bit difficult to follow through all the installation steps. There were a lot of precise steps to follow, and I was a bit nervous while doing it for fear of messing things up on my computer. I had to re-install Chocolatey because I forgot a step, but in the end, I could install everything correctly. The second challenge was importing the data, it took me a while to understand the Mongoimport function; the handshake was not happening, so it took me a while to figure it out. Ultimately, I understood how to write the path for importing my CSV file and that it required my connection string at the beginning of the code line to make the handshake happen. The third struggle was to connect my MongoDB profile with my code. It took me a while to realize I forgot to add my URI to the env file.

I made eight functioning queries with this dataset. The first one was to count the number of documents in the collection. I discovered there were 3599 documents in this collection. The creator of this dataset did not lie about the number mentioned in the description and title of this dataset, which was around 3400 videos collected.

In the second query, I asked to display all the collected data in the terminal. Only the first 100 documents are displayed in the terminal. The other 3499 documents were just mentioned at the end of the result. It has shown how heavy this dataset is.

The third query served to retrieve the data from the history category and only displayed their videos’ titles. I discovered 587 historical videos in this dataset, many of which were not English. I have noticed some of them had their tile in Hindi or Mandarin. It shows the potential variety of this dataset.

The fourth query was to find documents of videos with over 100 subscribers on their channel. I found some issues with the dataset through this query. Because the number of subscribers is written in the description (i.e., everything is written as a string in this input) and does not have its subcollection, I need to pull the numbers as a string from the description. Because of that, the results of the query were inaccurate. I did not receive an accurate number range when using comparison operators. Sometimes, the numbers do not match my request. For example, I found a lot of decimal numbers between 1 and 100, but those that were not in decimals were over 100.

I asked for a video with a specific ID for the fifth query. I looked through the data collection in the MongoDB Atlas, took an ID randomly, and asked for it in the code. It worked because I verified that the pulled data matched the document in my MongoDB Atlas.

For the sixth query, I filtered the art and music category by displaying only the videos with less than 400 subscribers on the channel, then giving only the title and its description in the results. As mentioned in the fourth query, the results were again inaccurate because the subscribers’ numbers are in a string and do not have their subcollection. Plus, the numbers are written with letters two, such as 90K or 1.1M, so it messes up the accuracy of the results. However, I discovered popular videos in this collection by seeing NSYNC’s Pop Music video as part of the results, which creates more variety in this dataset.

The seventh query was finding the food category videos, sorting them alphabetically, and limiting the results to five videos. It worked, and again, Japanese writings and the BuzzFeed video channel appeared in the results.

The last query was finding one video according to the input categories. I mixed categories that were not part of this dataset and some existing categories, and it could still pull one document that matched one of the existing categories.

Overall, I discovered that the used dataset for this assignment might not be a good fit for my final project, or at least very limited in its options. I tried various MongoDB functions, and only a few could pull data from this dataset. Plus, some fields in the documents do not allow me to pull the data. For example, if I couldn’t find a document by searching for a specific word within a string, I always received empty arrays or null as a result. So, if I want to search all the documents containing the word “India” in their titles, is impossible because the query is not precise enough, aka I would need to give a full title to receive something in the terminal. I think it is in how things were categorized; there are only a few categories and no subcollections in this dataset, limiting the possibilities of pulling out data. I also looked at the CSV file and noticed that some elements in the categories could have been categorized. For example, in the description category, many elements written in it could have been divided, such as the content creators’ names and the total of subscribers. The only category working well in this dataset is “category” because it is the only category that requires few keyword inputs for the MongoDB functions. All the other categories are too descriptive. For obscure reasons, I cannot insert the request alone in the find() function; I need to pass them first into a variable and use that variable in the function to make it work. If I want to use this dataset, I must edit it by creating subcollections in its categories to pull data effectively. However, I see that this dataset has a variety of collected videos, from the language to the popularity of the channels they’re coming from.