****

**NoSQL Assignment**

**By**

**Group 6**

**Mr. Wasin Heesawat 6288077**

**Ms. Intr-orn Lertsupakitsin 6288089**

**Mr. Krissanapong Palakham 6288102**

**Mr. Pongsakorn Piboonpongpun 6288107**

**Presented to**

**Dr. Wudhichart Sawangphol**

**A Report Submitted in Partial Fulfillment of**

**the Requirements for**

**ITCS495 Special Topics in Database and Intelligent Systems**

**Faculty of Information and Communication Technology**

**Mahidol University**

**Semester 1/2022**

Table of Contents

[I. Data Selection 3](#_Toc115437355)

[A. Purposes of data 3](#_Toc115437356)

[B. Data Model 3](#_Toc115437357)

[C. Motivation 3](#_Toc115437358)

[II. NoSQL Technology Selection 4](#_Toc115437359)

[A. Main Features 4](#_Toc115437360)

[B. Tools and Services 4](#_Toc115437361)

[III. NoSQL Implementation 7](#_Toc115437362)

[A. Database Creation 7](#_Toc115437363)

[C. Testing Data Queries 8](#_Toc115437364)

[D. NoSQL Service Information 10](#_Toc115437365)

[IV. References 11](#_Toc115437366)

# Data Selection

## Purposes of data

The purpose that we select this data is we see the insight of tv shows and movies on Netflix. For example, type, genres, and rating.

## Data Model

* + 1. **Show\_id** – This column contains the id of the show on Netflix (Format: stxxx).
    2. **Type –** This column contains the type of show on Netflix.
    3. **Title –** This column contains the title name of the show on Netflix.
    4. **Director –** This column contains the name of the director of each show on Netflix.
    5. **Cast –** This column contains the name of the actor or actress of each show on Netflix.
    6. **Country –** This column contains the country that produces the show which is published on Netflix.
    7. **Date\_added –** This column contains the date, month, and year of the show that was published on Netflix.
    8. **Realease\_year –** This column contains the date, month, and year of the show that was published on Netflix.
    9. **Rating –** This column contains the type of movie rating of each show on Netflix.
    10. **Duration –** This column contains the time duration of the show on Netflix.
    11. **Genres –** This column contains the genres of the show on Netflix.
    12. **Description –** This column contains the description of the show on Netflix.

## Motivation

We do it for people who like movies and want to see information about movies of interest such as who is the director or where the movie is made and can show users the type of movie they like or are interested in.

# NoSQL Technology Selection

## Main Features

The main feature used in this assignment is MongoDB technology. MongoDB is a well-known open-source NoSQL Database that provides a document-oriented database regarding JSON-like documents. [1] It can perform data without any data structures to query rather than RDBMS. In addition, MongoDB is suitable for hierarchical data storage that centers around the CAP theorem (Consistency, Availability, and Partition tolerance) and also has main features including [2]

* **Schema-less Database** - A schema-less database provides a collection that can keep different types of documents. To clarify easily, users could insert multiple types of data whether numbers of fields, content, and size. A document doesn’t have to be similar to another document like the RDBMS.
* **Scalability** - MongoDB can distribute data on multiple servers, so a large amount of data can be partitioned into data chunks using the Shard key. In addition, those data chunks are evenly distributed across Shards that reside across several Physical Servers.
* **Replication** - MongoDB provides high availability and redundancy by creating multiple copies of data and sending the copies to a different server to prevent one server fails, so users can receive data from another server.
* **Indexing** - Every field in the documents is indexed with primary and secondary indices, providing less time to get and search each document's data.

## Tools and Services

There are several tools, applications, and services that could be used to develop the NoSQL script. However, we came up with three technologies that are important and famous technologies providing built-in platforms, online document databases, and cloud services. The three technologies are included as follows:

* **NoSQL Booster for MongoDB**

[1] NoSQL Booster is the cross-platform tool with a graphical user interface (GUI) for MongoDB that contains multiple built-in including:

* + MongoDB Script Debugger
  + Comprehensive Server Monitoring Tools
  + Query Code Generator
  + Advanced IntelliSense Support

[3] NoSQL Booster admires its ability to automatically complete variables, methodologies, keywords, properties as well as collection names. In addition, it could write a script query faster and provides several features including:

* + **Chaining Fluent Query Interface** - NoSQL Booster provides a fluent query builder API that allows query construction using chaining syntax.
  + **Collection Re-Schema Tool** - The re-schema tool provides a GUI in order to update the MongoDB Collection Schema. It is essentially generating a code when users update by clicking only a few clicks.
  + **One-Click Grouping/Filtering** - The One-Click grouping feature allows grouping the selected field in query results to calculate counts, totals, maximums, minimums, and averages.
  + **Test Data Generator** - The Test Data Generator allows users to create random fake data for testing purposes. Because real data cannot always be distributed and restricted by the data’s owner, fake data can mimic it. So, users could ensure the statistically significant without any restrictions.
* **MongoDB Atlas**

The difficulty of installing, administering, and healing their deployments on the cloud service provider of their choice is handled by MongoDB Atlas, a fully managed cloud database (AWS, Azure, and GCP). The best approach to install, run, and scale MongoDB in the cloud is using the MongoDB Atlas. With Atlas, users can quickly and easily launch a MongoDB database with a few mouse clicks. [4] There are several features of MongoDB Atlas including: [5]

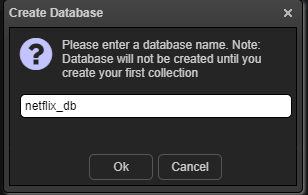
* + **Built-in Automation Features** – MongoDB Atlas has Automation tools for database administration that help users gain a competitive advantage. The automation features include as follows:
    1. Provisioning, setup, and configuration of hardware
    2. Upgrades and patches for software
    3. Disaster recovery and backups
  + **Easier Scalability –** Apps and systems must react to unanticipated needs and the company’s usual growth rate. MongoDB is a NoSQL database that allows for horizontal scalability via a technique known as Sharding.
  + **Strong Security Measures –** MongoDB Atlas employs a multilayered strategy. MongoDB Atlas has security features like authentication, authorization, encryption, and more. These features help prevent data breaches and improve security.
  + **Extensive Analytics and Monitoring –** MongoDB Atlas has built-in tracking and monitoring functions that provide valuable insights into the performance of your database.In addition, it gets real-time intuitions into its server clusters and set up personalized alerts, allowing users to fix issues before they affect performance or the end-user experience.
* **Google Cloud Platform (GCP)**

Google cloud platform or GCP is the public cloud that has various services such as Big Data, IoT, Data storage or etc. I can help the user develop the application or server that MongoDB can use Google cloud to make the easy work with data and take the operational heavy lifting away from the application teams and developers. [6] There are several features in GCP as follows:

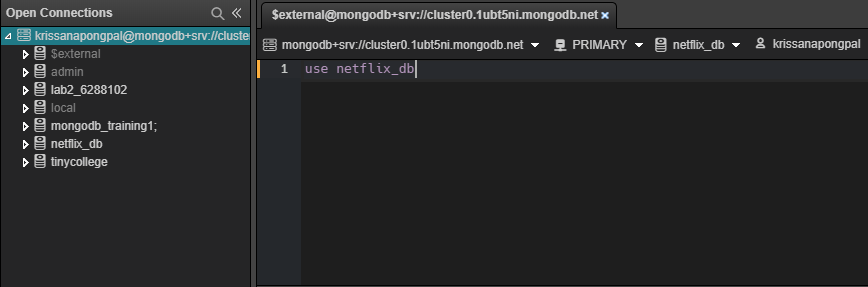
* + **Fault Tolerance and High Availability –** GCP makes sure that applications are not just consistently and conveniently accessible, but also very resilient to difficulties. The availability of GCP's services and products is also intended to assist with application deployment concerns.
  + **Scalability and Elasticity –** The built-in services of Google Cloud allow for automatic application scaling and load distribution. In addition, Users can automatically add or delete instances and set a policy that scales up or down the number of instances based on measured load and target utilization using Managed Instance Groups, a feature of GCP that allows you to manage identical instances as a single entity in a single zone.
  + **Disaster Recovery –** GCP has a robust global network, redundant points of presence throughout the world, a scalable service infrastructure, a highly secure network, and is generally privacy compliant, all of which help it meet an application's SLA needs.

# NoSQL Implementation

## Database Creation

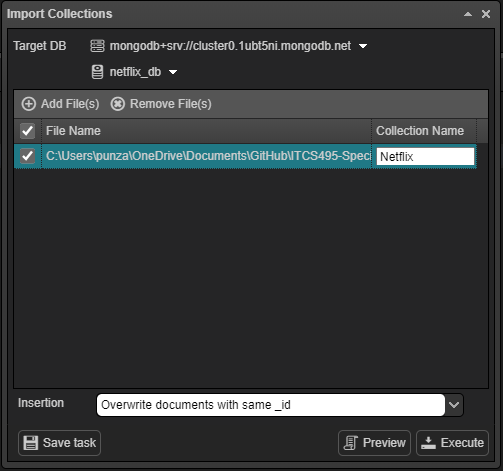
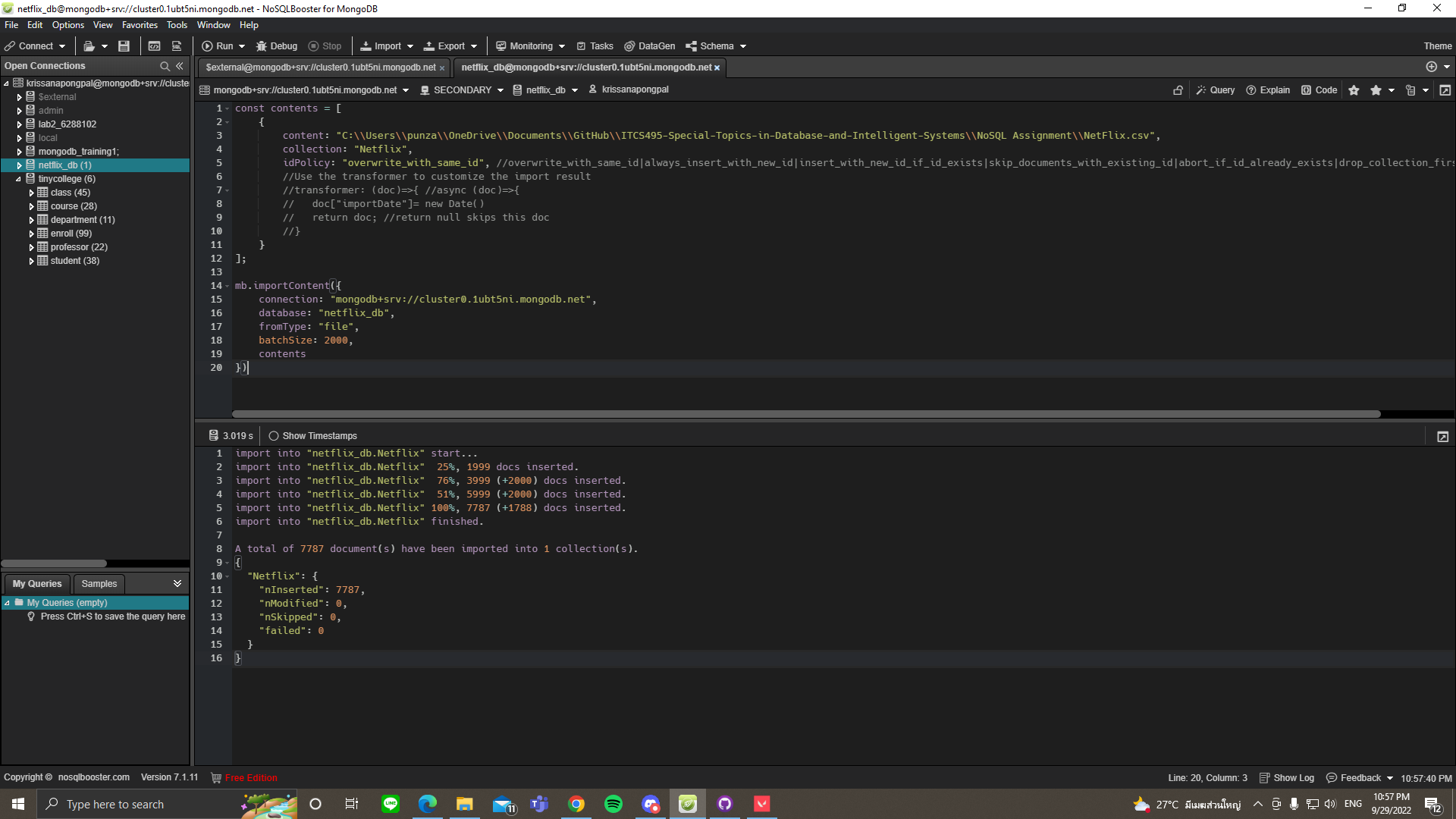
****

Firstly, we create the current database that we are working with. We will set the name of the database as “netflix\_db”. After that, we click the “Ok” button, the MongoDB will create the database.



After MongoDB creates the database, the database will show on the left of the program. Then, we call to use the "netflix\_db" command to use the database we are working with. Finally, we will change the database type to “netflix\_db”.

* 1. **Data Import**

** **

****

After we create the database, we import the collection name “Netflix” to “netflix\_db” as a target DB and select “Overwrite documents with same\_id” for the insertion. Finally, select the execute to show up in the Netflix collection and continuously work with it.

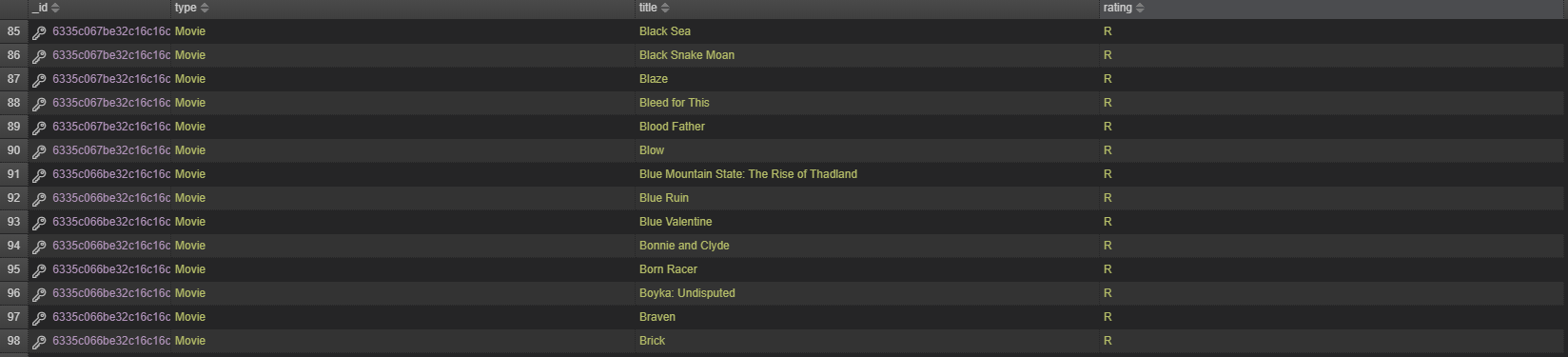
## Testing Data Queries

* **Query the Restricted movies that show title, type, rating, and title show in ascending order**

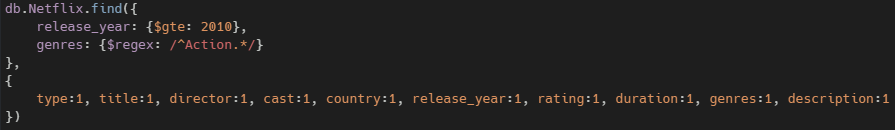


For the first query, we want to find the restricted movies and show the title, type, rating, and title show in ascending order. We write a find command that the rating is ‘R’ and use the projection to specify the title, type, and rating. We add a “sort function” to sort the tile in ascending order.

**Result:**



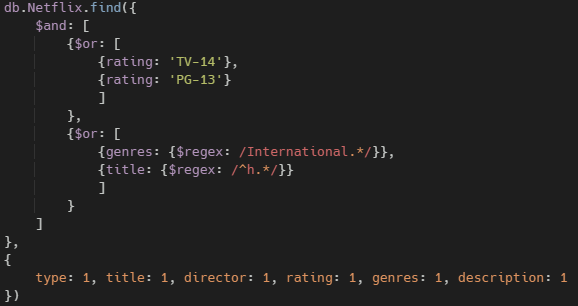
* **Query movies after or equal to 2010 that show type, title, director, cast, country, rating, duration, genres, description, and genres is action.**



For the second query, we want to find the action movies that have the release year after 2010 and show the result as type, title, director, cast, country, rating, duration, genres, and description. We write a find command that release\_year is greater than or equal to 2010 and genres use regex for find pattern matching that we find the pattern is “Action” and use the projection to specify type, title, director, cast, country, rating, duration, genres, description.

**Result: **

* **Query movies’ type, title, director, rating, genres, and description that rating is TV-14 or PG-13, also genre is international both movies and TV shows, or title contains “S” at the first letter**



The third query is created in order to find the Netflix movies and TV shows that rating “TV-14” or “PG-13” representing content for teens. Moreover, the query not only finds a rating but also contains “International” genres or titles containing “H” at the first letter. We start by using “$and” in order to retrieve both ratings and others. In the first code for “$and”, we use “$or” in order to choose ratings for both “TV-14” and “PG-13”. [7] In the second code for “$and”, we also use “$or” in order to choose genres that contain “International” or title that contains “H” at the first letter by using “$regex” for both scripts. In addition, we also conduct the projection that we set it to 1 in order to represent an only type, title, director, rating, genres, and description from queries.

**Result:**



## NoSQL Service Information

We conduct this assignment by using the MongoDB services. However, in order to access the MongoDB databases, users are required to access via NoSQL Booster for MongoDB by following these steps:

1. Open the NoSQL Booster for MongoDB on your computer
2. Open a connection and connect from URI and insert “mongodb+srv://krissanapongpal:za4DwPWLW560O13X@[cluster0.1ubt5ni.mongodb.net/test](http://cluster0.1ubt5ni.mongodb.net/test)”
3. Using a database named “netflix\_db” and collection “Netflix” in order to retrieve a data

# References

|  |  |
| --- | --- |
| [1] | R. Tiwari, "NoSQLBooster For MongoDB Simplified: A Comprehensive Guide 101," 16 February 2022. [Online]. Available: https://hevodata.com/learn/nosqlbooster/#con. [Accessed 29 September 2022]. |
| [2] | A. Saini, "What is MongoDB – Working and Features," Geeksforgeeks, 6 June 2021. [Online]. Available: https://www.geeksforgeeks.org/what-is-mongodb-working-and-features/. [Accessed 29 September 2022]. |
| [3] | Scalegrid, "Which is the Best MongoDB GUI?," Scalegrid, 15 August 2019. [Online]. Available: https://scalegrid.io/blog/which-is-the-best-mongodb-gui/. [Accessed 29 September 2022]. |
| [4] | MongoDB, "MongoDB Atlas Tutorial," MongoDB, [Online]. Available: https://www.mongodb.com/basics/mongodb-atlas-tutorial. [Accessed 29 September 2022]. |
| [5] | Abraham, "Top 10 features of MongoDB Atlas," Fosslinux, 27 January 2022. [Online]. Available: https://www.fosslinux.com/50299/features-of-mongodb-atlas.htm. [Accessed 29 September 2022]. |
| [6] | MongoDB, "MongoDB on Google Cloud," MongoDB, [Online]. Available: https://www.mongodb.com/mongodb-on-google-cloud. [Accessed 29 September 2022]. |
| [7] | Netflix, "Maturity ratings for TV shows and movies on Netflix," Netflix, [Online]. Available: https://help.netflix.com/th/node/2064/us. [Accessed 29 September 2022]. |