**PART 2 - READING ASSIGNMENT (Papers Attached)**

1. Application of NoSQL Database in Web Crawling

In web crawling, because of the huge amount of information, the performance of search engine is mostly affected by storage form and the storage database. The author of this paper compare the performance of relational database with MongoDB (NoSQL) by using Meteorological BBS information collection system.

The result is that relational database has multiple tables’ storage with foreign key, sharp decline in query performance with huge amount of data, and vertical scalability with high cost. But MongoDB supports schema-free, has great query performance with huge amount of data and provides easy horizontal scalability with low cost of hardware. It is more suitable for data storage in Web crawling.

1. Comparing NoSQL MongoDB to an SQL

In this paper, the author compares the performance of MongoDB with Microsoft SQL Server Express by many kinds of test case - Insertion portion, update portion, and other different select queries. As a result, the author find that MongoDB has better runtime performance for inserts, updates and simple queries. SQL performed better when updating and querying non-key attributes, as well as for aggregate queries.

As for the conclusion, the author recommends MongoDB as a good solution for larger data sets in which the schema is constantly changing or in the case that queries performed will be less complex. And For those users that have a strict schema defined and a modest amount of structured data. Although they also found MongoDB to perform better than SQL in general, MongoDB has poor performance for aggregate functions and querying based on non-key values. Also, MongoDB required additional effort in implementation compared to SQL and required decisions that affected its performance. Lastly, SQL is the industry standard and much more widely supported over MongoDB.

1. Data Aggregation System

In this report, the author shows us a system called DAS (Data Aggregation System) which is used for information retrieval. He introduces the system from three aspect architecture, implementation and benchmark. Firstly, the architecture of DAS was designed as a series of independent components, which can scale from all running on a single node to multiple nodes and duplicate components. Secondly, he explains implementation from the Query Language it used, the services, the Caching and merging of data in DAS and its analytics system. Finally, he benchmarks DAS performance and result shows the average latency for fetching a single randomly-selected document from a wildcard query, with a given number of parallel clients.

One important things about DAS which is related to this course is that the cache of DAS consists of one or more MongoDB shards. This happens because MongoDB is a document store which natively stores the JSON (Javascript Object Notation) documents, which DAS uses as its internal representation.

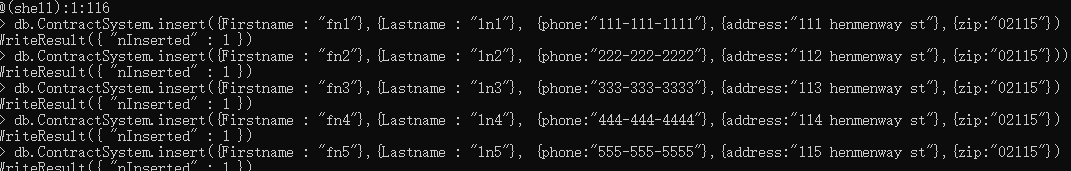
And the choice of a document store instead of a traditional relational database was driven by the need to be able to store and subsequently query individual members of the deeply-nested data structures that DAS handles, the structure of which are not known until run-time.

**PART 3 - PROGRAMMING ASSIGNMENT**

1. **Create database “userdb”**

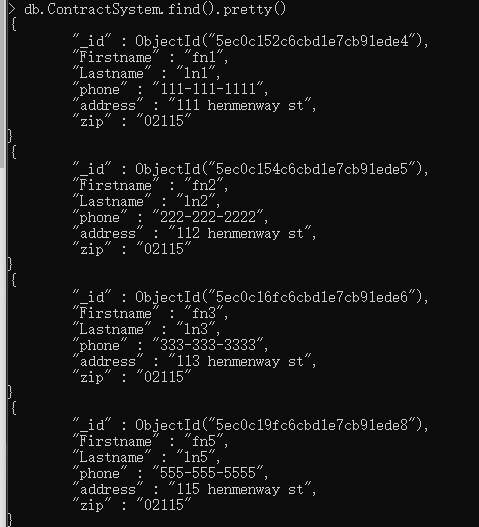


1. **Create collection “ContractSystem” and insert 5 document.**



1. **Delete one document**



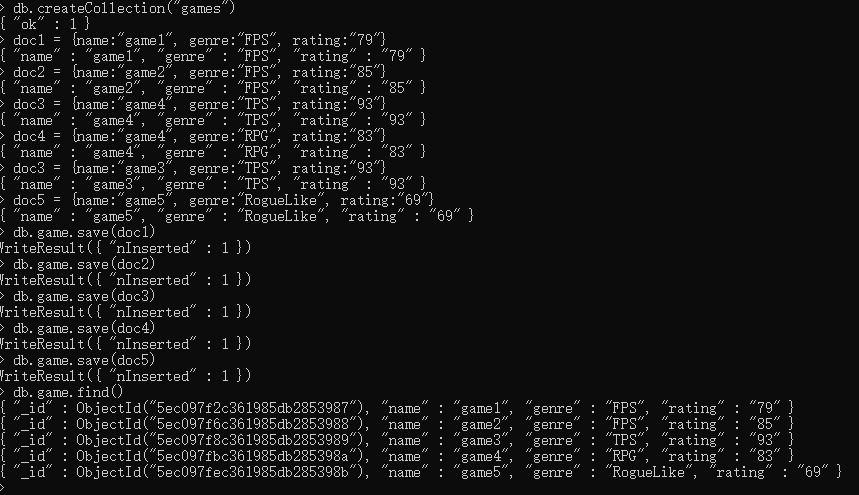


1. **Update one record**



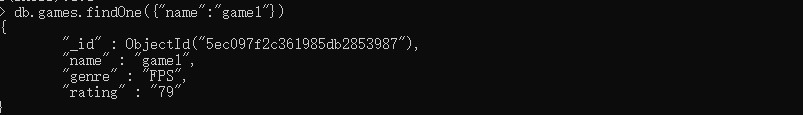
**PART 4 - PROGRAMMING ASSIGNMENT**

1. **Create new collection and documents. And a query that returns all the games.**

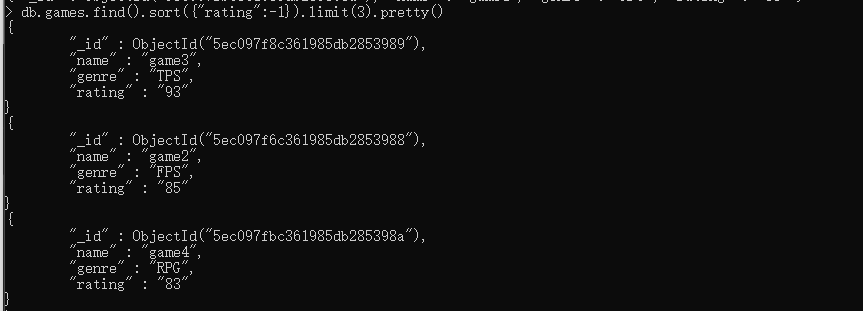


1. **Query to find one of games by name without using limit().**





1. **Query to find three highest rated games.**



1. **Update your two favorite games to have two achievements called ‘Game Master’ and ‘Speed Demon.**

**First way**

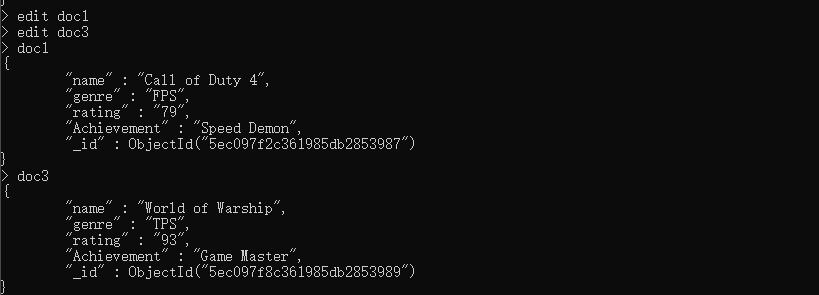




**For question 5:**



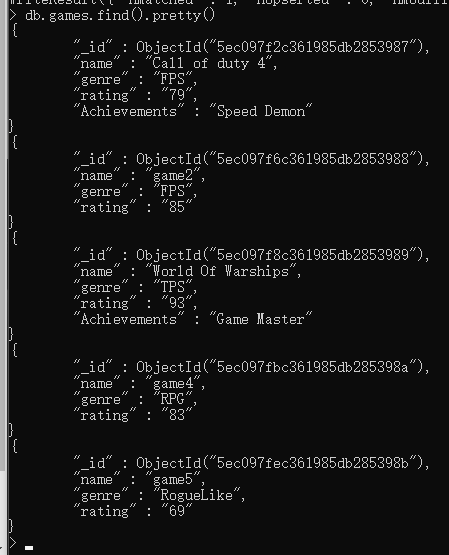
**Second way**



Missed code：db.games.save(doc1/doc3)

**And**

**Result**



1. **A query that returns all the games that have both the ‘Game Master’ and the ‘Speed Demon’ achievements.**

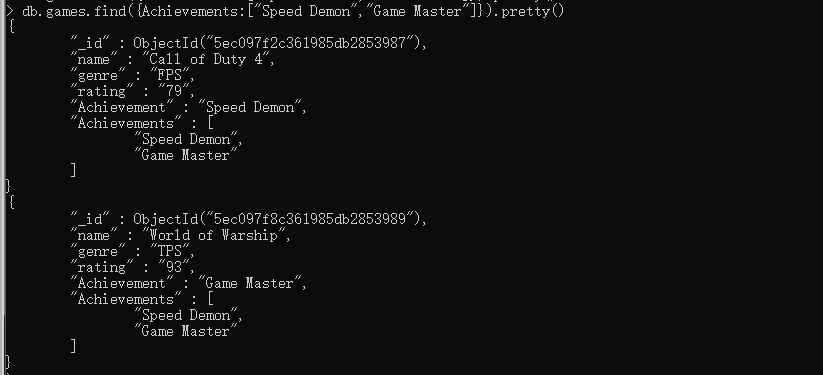




Igonre the key named “Achievement“ for last query

This query will only use key “Achievements





1. **query that returns only games that have achievements.**



