**MongoDB**

**Data Science Masters Assignment**

**17th Feb 2023**

**Q1. What is MongoDB? Explain non-relational databases in short. In which scenarios it is preferred to use MongoDB over SQL databases?**

**Sol.** MongoDB is an open source NoSQL database management program. NoSQL (Not only SQL) is used as an alternative to traditional relational databases. NoSQL databases are quite useful for working with large sets of distributed data. MongoDB is a tool that can manage document-oriented information, store or retrieve information.

MongoDB is used for high-volume data storage, helping organizations store large amounts of data while still performing rapidly. Organizations also use MongoDB for its ad-hoc queries, indexing, load balancing, aggregation, server-side JavaScript execution and other features.

A non-relational database is a database that does not use the tabular schema of rows and columns found in most traditional database systems. Instead, non-relational databases use a storage model that is optimized for the specific requirements of the type of data being stored.

**Scalability**: MongoDB is horizontally scalable, which helps reduce the workload and scale your business with ease.

**Manageability**: The database doesn’t require a database administrator. Since it is fairly user-friendly in this way, it can be used by both developers and administrators.

**Q2. State and Explain the features of MongoDB.**

**Sol.** These are some important features of MongoDB:

**1. Support ad hoc queries**

In MongoDB, you can search by field, range query and it also supports regular expression searches.

**2. Indexing**

You can index any field in a document.

**3. Replication**

MongoDB supports Master Slave replication.

A master can perform Reads and Writes and a Slave copies data from the master and can only be used for reads or back up (not writes)

**4. Duplication of data**

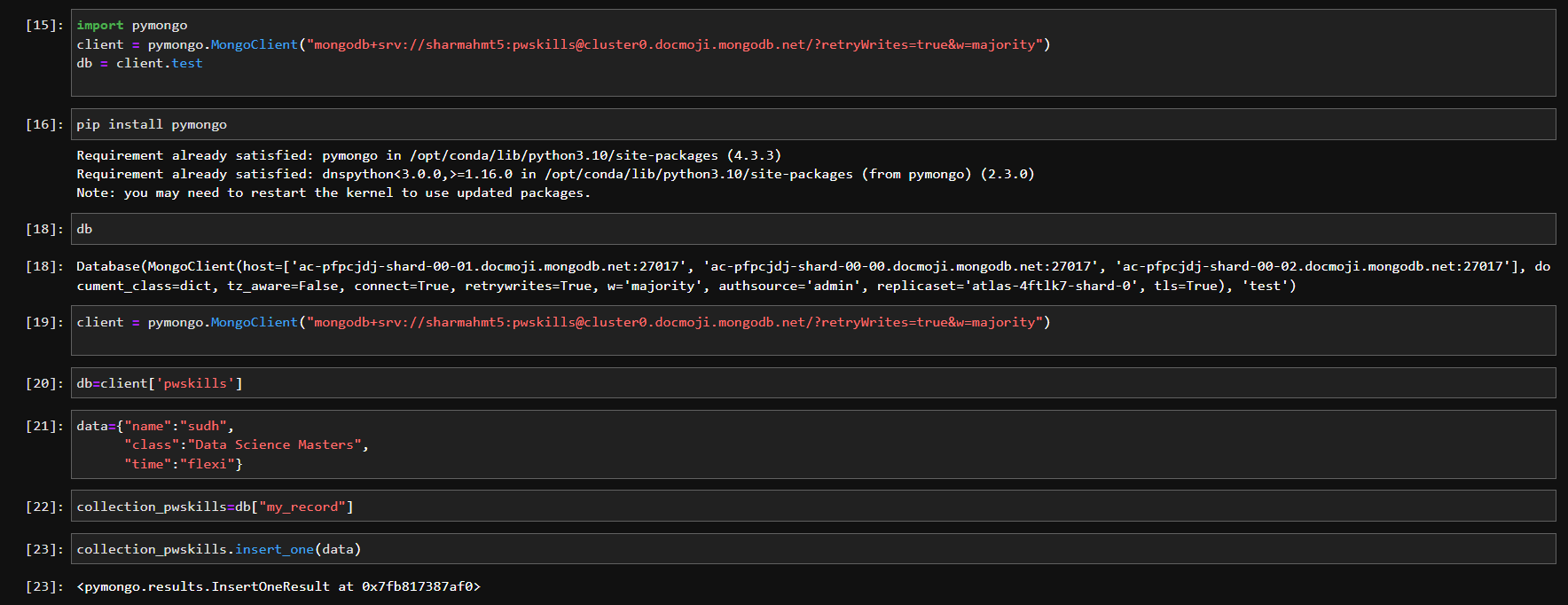
MongoDB can run over multiple servers. The data is duplicated to keep the system up and also keep its running condition in case of hardware failure.

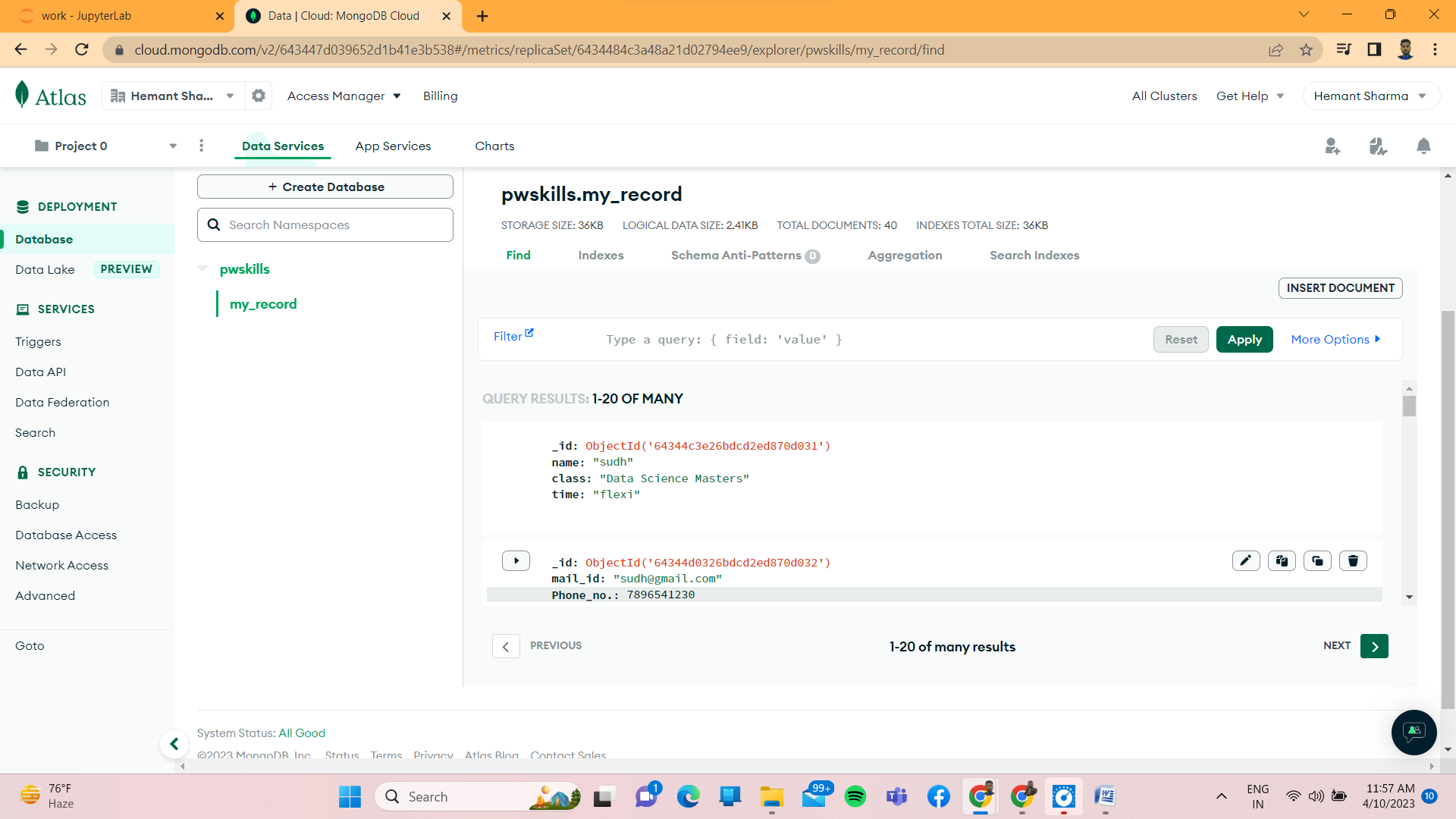
**5. Load balancing**

It has an automatic load balancing configuration because of data placed in shards.

**Q3. Write a code to connect MongoDB to Python. Also, create a database and a collection in MongoDB.**

**Sol.**

****



**Q4. Using the database and the collection created in question number 3, write a code to insert one record, and insert many records. Use the find() and find\_one() methods to print the inserted record.**

**Sol. *For insert\_one***

data1={"name":"sudh",

"class":"data science masters",

"time":"flexible"}

collection\_pwskills=db["my\_record"]

collection\_pwskills.insert\_one(data1)

***For Insert\_many***

data1={"name":"sudh",

"class":"data science masters",

"time":"flexible",

"name":"krish",

"class":"data science masters",

"time":"flexible",

"name":"hmt",

"class":"data science masters",

"time":"flexible"}

collection\_pwskills=db["my\_record"]

collection\_pwskills.insert\_many(data1)

***Use of Find()***

data1={"name":"sudh",

"class":"data science masters",

"time":"flexible",

"name":"krish",

"class":"data science masters",

"time":"flexible",

"name":"hmt",

"class":"data science masters",

"time":"flexible"}

collection\_pwskills=db["my\_record"]

collection\_pwskills.find({"name":"sudh"})

**Q5. Explain how you can use the find() method to query the MongoDB database. Write a simple code to demonstrate this.**

**Sol.** { "\_id": "apples", "qty": 5 }

{ "\_id": "bananas", "qty": 7 }

{ "\_id": "oranges", "qty": { "in stock": 8, "ordered": 12 } }

{ "\_id": "avocados", "qty": "fourteen" }

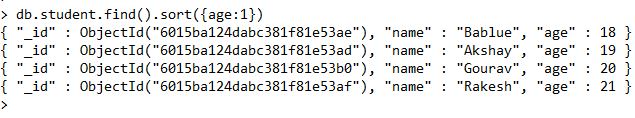
db.collection.find( { qty: { $gt: 4 } } )

**Q6. Explain the sort() method. Give an example to demonstrate sorting in MongoDB.**

**Sol.**

****

db.student.find().sort({age:1})



Q7. Explain why delete\_one(), delete\_many(), and drop() is used.

Sol.

**For delete\_one()**

data={“address”:”mountain 21”}

mycol.delete\_one(data)

**For delete\_many()**

data={“address”:{”$regex”:”^S”}}

mycol.delete\_many(data)