**MongoDB**

**Assignment - 3**

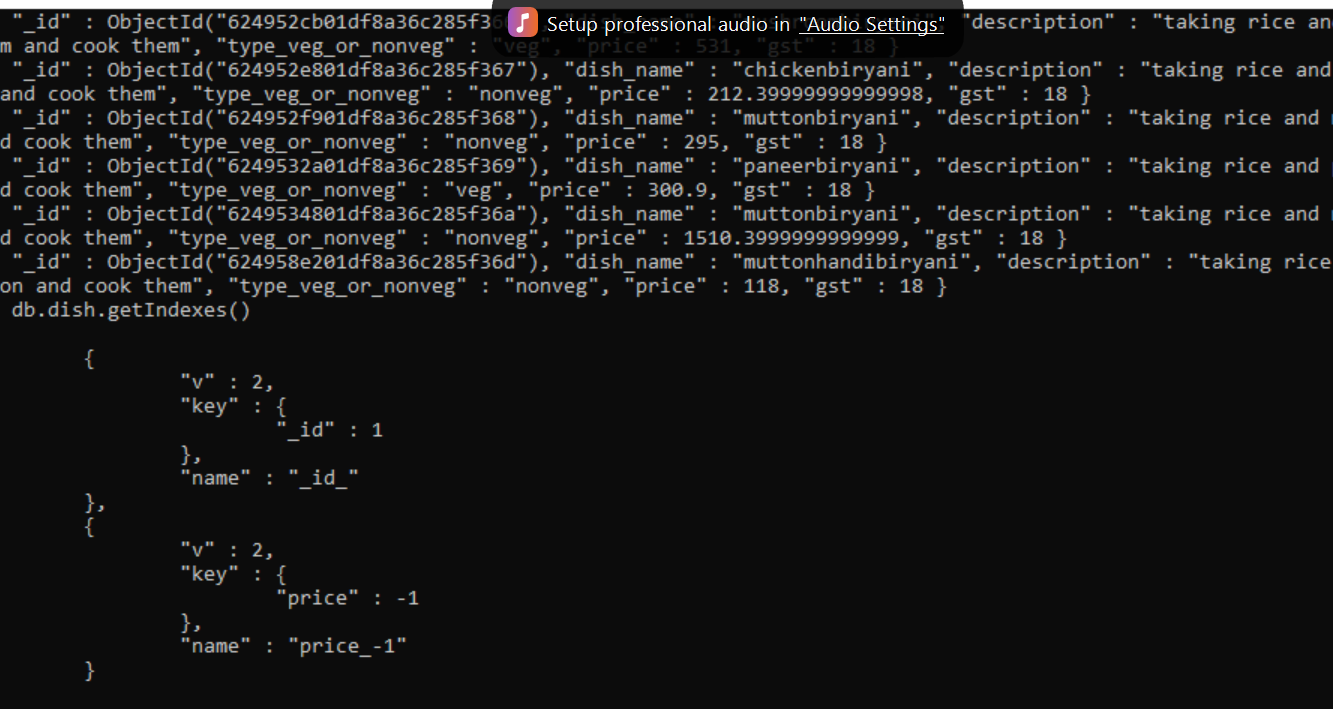
Create collection which creates objects ,use insertOne() & insertMany() methods for those objects.

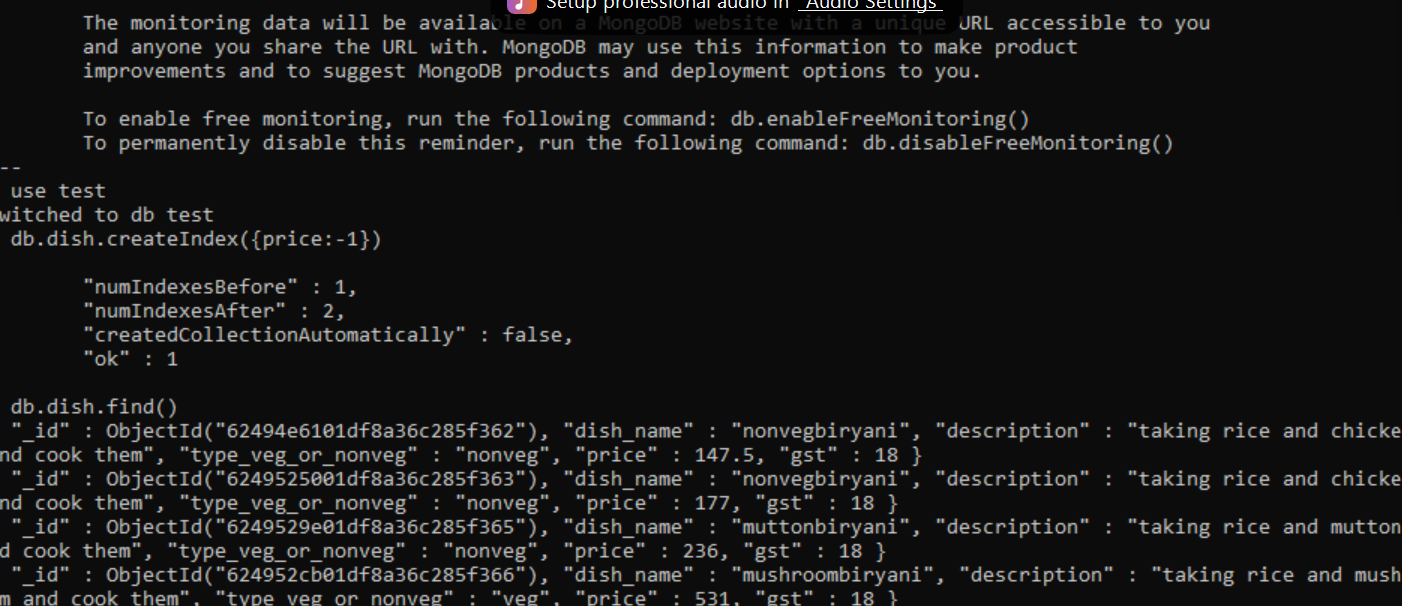
Create collection which uses find() and finfOne() methods for finding created documents.also use sort(),Limit() and ship() methods for the documents created."

Answer :

1. Use index for frequent opeations.

Create Replica Set on the same and check if it is created correctly.





Step 1) Ensure all mongod.exe instances added

Ensure that all mongod.exe instances which will be added to the replica set are installed on different servers. This is to ensure that even if one server goes down, the others will be available and hence other instances of MongoDB will be available.

Step 2) All mongo.exe instances connect to each other

Ensure that all mongo.exe instances can connect to each other. From ServerA, issue the below 2 commands.

mongo –host ServerB –port 27017

mongo –host ServerC –port 27017

Similarly, do the same thing from the remaining servers.

Step 3) Start the first mongod.exe instance

Start the first mongod.exe instance with the replSet option. This option provides a grouping for all servers which will be part of this replica set.

mongo –replSet "Replica1"

Where “Replica1” is the name of your replica set. You can choose any meaningful name for your replica set name.

Step 4) First server is added to the replica set

Now that the first server is added to the replica set, the next step is to initiate the replica set by issuing the following command rs.initiate ().

Step 5) Verify the replica set

Verify the replica set by issuing the command rs.conf() to ensure the replica set up properly.

Replica Set: Adding a Secondary using rs.add()

The secondary servers can be added to the replica set by just using the rs.add command. This command takes in the name of the secondary servers and adds the servers to the replication set.

Step 1) Suppose if you have ServerA, ServerB, and ServerC, which are required to be part of your replica set and ServerA, is defined as the primary server in the replica set.

To add ServerB and ServerC to the replica set issue the commands

rs.add("ServerB")

rs.add("ServerC")

Replica Set: Reconfiguring or Removing using rs.remove()

To remove a server from the configuration set, we need to use the “rs.remove” command

Step 1) First perform a shutdown of the instance which you want to remove. One can do this by issuing the db.shutdownserver command from the mongo shell.

Step 2) Connect to the primary server

Step 3) Use the rs.remove command to remove the required server from the replica set. So suppose

if you have a replica set with ServerA, ServerB, and ServerC, and you want to remove ServerC from the replica set, issue the command rs.remove("ServerC")