mongoDB

INSTALLATION GUIDE FOR MONGODB:

Here is a brief installation guide for MongoDB shell, MongoDB GUI, and MongoDB Community Server:

1. **MongoDB Shell (mongosh)**:

- Visit the MongoDB Download Center: https://www.mongodb.com/try/download/shell
- Select your operating system and download the appropriate installer.
- Follow the installation instructions provided for your specific operating system.
- Once installed, you can run `mongosh` in your terminal to start the MongoDB shell.

2. **MongoDB GUI (MongoDB Compass)**:

- Visit the MongoDB Download Center: https://www.mongodb.com/try/download/compass
- Choose your operating system and download the MongoDB Compass installer.
- Follow the on-screen instructions to install MongoDB Compass.
- After installation, launch MongoDB Compass to connect to your MongoDB databases using a graphical user interface.

3. **MongoDB Community Server**:

-Visit the MongoDB Download Center: https://www.mongodb.com/try/download/community
- Select your operating system and download the MongoDB Community Server installer.
- Follow the installation instructions provided for your specific operating system.
- Once installed, start the MongoDB server using the appropriate command for your OS.
- You can connect to the MongoDB server using the MongoDB shell or a GUI tool like MongoDB Compass.

INTRODUCTION

WHAT IS NOSQL?

NoSQL stands for "Not Only SQL." It refers to a category of databases that are designed to handle large volumes of data, diverse data types, and high-velocity data. Unlike traditional relational databases, NoSQL databases do not use tables with rows and columns. Instead, they use different data models such as documents, key-value pairs, wide-columns, or graphs. This flexibility allows them to efficiently handle big data and real-time web applications.

WHAT IS MONGODB?

MongoDB is a type of NoSQL database. It is a document-oriented database, which means it stores data in flexible, JSON-like documents. These documents can have different structures, allowing you to store and manage data more dynamically and efficiently.

KEY CONCEPTS OF MONGODB:

DOCUMENT

The basic unit of data in MongoDB. Documents are like records in a relational database but more flexible. Each document is a JSON-like structure, meaning it is easy to read and write.

Characteristics of a Document:
Schema-less: Unlike relational databases, documents in a collection do not need to have the same set of fields or structure. This allows for flexible and dynamic schemas.
Key-Value Pairs: A document consists of key-value pairs, where keys are strings, and values can be a variety of data types, including other documents, arrays, and arrays of documents.
Nested Documents: Documents can contain nested documents, which allows for more complex data representations.
Unique Identifier: Each document has a unique _id field, which acts as the primary key.

Here is a simple example of a MongoDB document:

```
{
  "name": "Isaac Clark",
  "age": 22,
  "courses": ["English", "Creative Writing", "Film Studies"],
  "gpa": 3.7,
  "home_city": "San Jose",
  "blood_group": "A-",
  "is_hotel_resident": false
},
{
  "name": "Jessica Moore",
  "age": 19,
  "courses": ["Biology", "Ecology", "Marine Science"],
  "gpa": 3.1,
  "home_city": "Austin",
  "blood_group": "B-",
  "is_hotel_resident": true
},
```

DATABASE:

A database is a container for collections, which are analogous to tables in a relational database.In MongoDB, the use database_name command is used to switch to a specific database.
When you execute the use database_name command, you are setting the context for subsequent operations.
If the specified database does not exist, MongoDB will create it for you when you first insert data into a collection within that database. Simply using the use database_name command alone does not create the database.
By specifying the database once using the use command, you do not have to include the database name in every command. This simplifies your operations and makes your commands more readable.

COLLECTIONS :

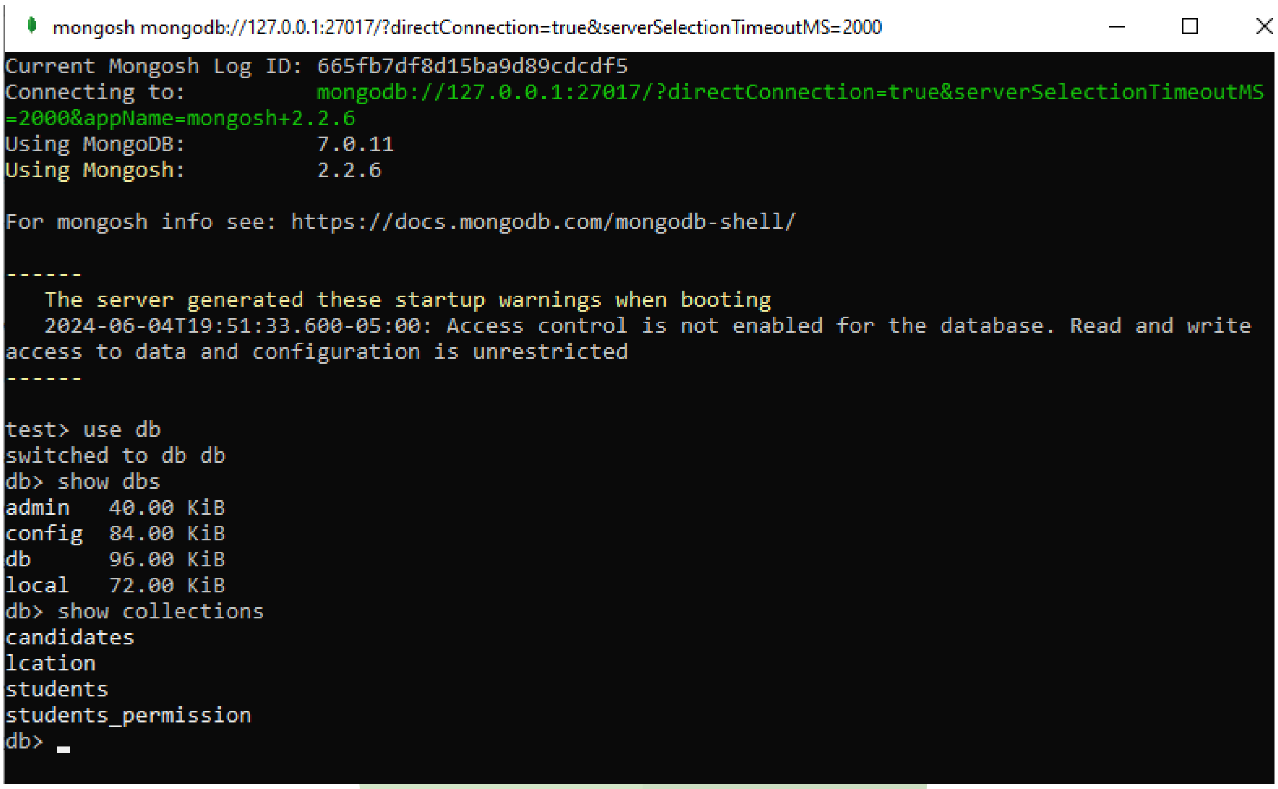
MongoDB stores documents in collections. Collections are analogous to tables in relational databases.
If a collection does not exist, MongoDB creates the collection when you first store data for that collection. Both the insertOne() and the createIndex() operations create their respective collection if they do not already exist. Be sure that the collection name follows MongoDB Naming Restrictions.
MongoDB provides the db.createCollection() method to explicitly create a collection with various options, such as setting the maximum size or the documentation validation rules. If you are not specifying these options, you do not need to explicitly create the collection since MongoDB creates new collections when you first store data for the collections.
To obtain a list of MongoDB collections, we need to use the Mongo shell command show collections. This command will return all collections created within a MongoDB database. To be able to use the command, we'll first need to select a database where at least one collection is stored.

FIELD

A field is a key-value pair in a document. The key is a string, and the value can be of various data types, including arrays and nested documents.MongoDB stores underlying document data using BSON types, and Mongoid converts BSON types to Ruby types at runtime in your application. For example, a field defined with type: :float will use the Ruby Float class in-memory and will persist in the database as the the BSON double type.

Field type definitions determine how Mongoid behaves when constructing queries and retrieving/writing fields from/to the database.

The snapshot provided shows the execution of MongoDB shell commands: switching database with ‘use db’,displaying collections with ‘show collections’ and listing database with ‘show dbs’.



```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
Current Mongosh Log ID: 665fb7df8d15ba9d89cdcdf5
Connecting to:          mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS
=2000&appName=mongosh+2.2.6
Using MongoDB:          7.0.11
Using Mongosh:          2.2.6

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

------
   The server generated these startup warnings when booting
   2024-06-04T19:51:33.600-05:00: Access control is not enabled for the database. Read and write
access to data and configuration is unrestricted
------

test> use db
switched to db db
db> show dbs
admin   40.00 KiB
config  84.00 KiB
db      96.00 KiB
local   72.00 KiB
db> show collections
candidates
lcation
students
students_permission
db> _
```