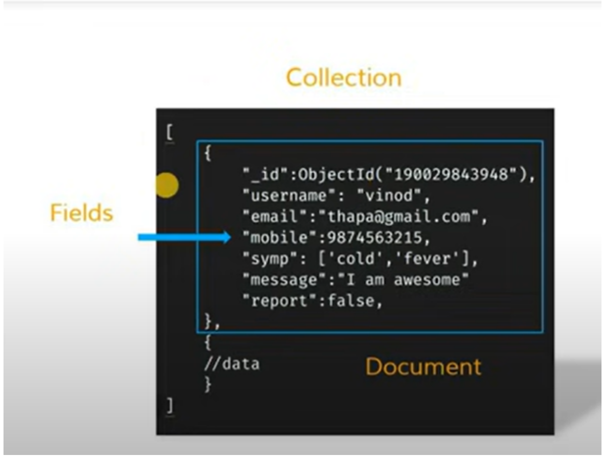
MongoDB

- It is a Non -relational database management system works on document-based Database

- It stores data as collections in the BSON format

  (JSON like structure/format) - [JSON vs BSON | Find Out The 10 Useful Differences]

- BSON is a binary representation to store data in JSON format

- 

```
[
    {
        "_id":ObjectId("190029843948"),
        "username": "vinod",
        "email":"thapa@gmail.com",
        "mobile":9874563215,
        "symp": ['cold','fever'],
        "message":"I am awesome"
        "report":false,
    },
    {
    //data
    }
]
```

- Provides flexibility of Number of fields irrespective of document

- Installation : 🔴 #2: Complete Installation of MongoDB | Setting Environment ...

- GUI Installation : 😍 #8: Install MongoDB GUI Compass in Hindi in 2020

- Default Port Number
  - Mongo : **27017**
  - SQL : **3306**

- Version
    - Mongo shell : mongo - - version
    - MongoDB : mongod - - version

Commands In Mongo Shell

- **mongo** : used to start mongo shell

- **cls** : used to clean the screen

- **show dbs** : used to list dbs that has at least one collection

- **Use db_name** : used to create a database(**USE EVERY TIME**)

- **db** : used to list the current active database.

- **show collections** : used to list collections that has at least one document **(Enter "use db_name" before using this command)**

- **db.collection_name.insertOne({** name : "NANI" , id : "1" **})**
    - used to create one document of specified db and collection
    - Id gets Created by default if not declared

- **db.collection_name.Many([{},{},{}])**
    - used to create documents of specified db and collection

- **db.collection_name.find()**
    - used to list all documents of specified db and collection

- **db.collection_name.find().pretty()**
    - **.pretty()** is used lists in a pretty manner

- **db.collection_name.find(query,projection)**
    - Ex: db.collection.find({“name”: “NANI”})
        - displays all fields of that particular document
    - Ex: db.collection.find({“name”: “NANI”},{name : 0})
        - displays only id, name of that particular document
    - Ex: db.collection.find({“name”: “NANI”},{name : 1})
        - displays all fields of that particular document
    - Ex: db.collection.find({“name”: “NANI”},{ _id:0,name : 1})
        - displays only name of that particular document

- **db.collection_name.find(query,projection).limit(n)**
    - displays only n number of fields satisfying the query
    - If n == 1
        - **db_name.collection_name.findOne(query,projection)**

- **db.collection_name.find(query,projection).limit(n).skip(m)**
    - displays only n number of fields satisfying the query skipping m fields

- **db.collection_name.updateMany(<filter>,<update>)**
    - If n == 1
        - **db_name.collection_name.updateOne(<filter>,<update>)**
    - Ex: db.collection.updateOne({“name”: “NANI”}, {$set: {“Gender”: “MALE”}})
        - displays all fields of that particular document
    - The **$set** Operator replaces the value of a field with specified value

- **db.collection_name.deleteMany(DELETION_CRITERIA)**
    - Ex: db.collection.deleteMany({“Gender”: “MALE”})
        - Deletes all the documents having the criteria
    - Ex: db.collection.deleteMany({})
        - Deletes all the documents
    - What's the difference between "deletemany" and "remove" in mongodb? - Stack Overflow

- **Ctrl + c / quit() :** used to quit from the mongo shell